```
(pid=24022)         'agent_1': np.ndarray((2,), dtype=int64, min=9.0, max=9.0,
mean=9.0)},
(pid=24022)    2: { 'agent_0': np.ndarray((2,), dtype=int64, min=10.0, max=10.0,
mean=10.0),
(pid=24022)         'agent_1': np.ndarray((2,), dtype=int64, min=10.0, max=10.0,
mean=10.0)},
(pid=24022)    3: { 'agent_0': np.ndarray((2,), dtype=int64, min=3.0, max=3.0,
mean=3.0),
(pid=24022)         'agent_1': np.ndarray((2,), dtype=int64, min=3.0, max=3.0,
mean=3.0)},
(pid=24022)    4: { 'agent_0': np.ndarray((2,), dtype=int64, min=9.0, max=9.0,
mean=9.0),
(pid=24022)         'agent_1': np.ndarray((2,), dtype=int64, min=9.0, max=9.0,
mean=9.0)},
(pid=24022)    5: { 'agent_0': np.ndarray((2,), dtype=int64, min=4.0, max=4.0,
mean=4.0),
(pid=24022)         'agent_1': np.ndarray((2,), dtype=int64, min=4.0, max=4.0,
mean=4.0)},
(pid=24022)    6: { 'agent_0': np.ndarray((2,), dtype=int64, min=4.0, max=4.0,
mean=4.0),
(pid=24022)         'agent_1': np.ndarray((2,), dtype=int64, min=4.0, max=4.0,
mean=4.0)},
(pid=24022)    7: { 'agent_0': np.ndarray((2,), dtype=int64, min=10.0, max=10.0,
mean=10.0),
(pid=24022)         'agent_1': np.ndarray((2,), dtype=int64, min=10.0, max=10.0,
mean=10.0)}}
(pid=24022) 2019-10-05 12:10:22,203        INFO sampler.py:305 -- Info return from
env: { 0: {'agent_0': {}, 'agent_1': {}},
(pid=24022)    1: {'agent_0': {}, 'agent_1': {}},
(pid=24022)    2: {'agent_0': {}, 'agent_1': {}},
(pid=24022)    3: {'agent_0': {}, 'agent_1': {}},
(pid=24022)    4: {'agent_0': {}, 'agent_1': {}},
(pid=24022)    5: {'agent_0': {}, 'agent_1': {}},
(pid=24022)    6: {'agent_0': {}, 'agent_1': {}},
(pid=24022)    7: {'agent_0': {}, 'agent_1': {}}}
(pid=24022) 2019-10-05 12:10:22,203        INFO sampler.py:403 -- Preprocessed
obs: np.ndarray((2,), dtype=int64, min=0.0, max=0.0, mean=0.0)
(pid=24022) 2019-10-05 12:10:22,204        INFO sampler.py:407 -- Filtered obs:
np.ndarray((2,), dtype=float64, min=0.0, max=0.0, mean=0.0)
(pid=24022) 2019-10-05 12:10:22,212        INFO sampler.py:521 -- Inputs to
compute_actions():
(pid=24022)
(pid=24022) { 'agent_0': [ { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 0,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                            'prev_reward': 0.0,
(pid=24022)                            'rnn_state': []},
(pid=24022)                  'type': 'PolicyEvalData'},
(pid=24022)                { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 1,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=0.707, max=0.707, mean=0.707),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
```

```
(pid=24022)                                 'prev_reward': 0.0,
(pid=24022)                                 'rnn_state': []},
(pid=24022)                   'type': 'PolicyEvalData'},
(pid=24022)                { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 2,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=0.666, max=0.666, mean=0.666),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                            'prev_reward': 0.0,
(pid=24022)                            'rnn_state': []},
(pid=24022)                   'type': 'PolicyEvalData'},
(pid=24022)                { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 3,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.521, max=-0.521, mean=-0.521),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                            'prev_reward': 0.0,
(pid=24022)                            'rnn_state': []},
(pid=24022)                   'type': 'PolicyEvalData'},
(pid=24022)                { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 4,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=0.631, max=0.631, mean=0.631),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                            'prev_reward': 0.0,
(pid=24022)                            'rnn_state': []},
(pid=24022)                   'type': 'PolicyEvalData'},
(pid=24022)                { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 5,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.45, max=-0.45, mean=-0.45),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                            'prev_reward': 0.0,
(pid=24022)                            'rnn_state': []},
(pid=24022)                   'type': 'PolicyEvalData'},
(pid=24022)                { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 6,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.416, max=-0.416, mean=-0.416),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                            'prev_reward': 0.0,
(pid=24022)                            'rnn_state': []},
(pid=24022)                   'type': 'PolicyEvalData'},
(pid=24022)                { 'data': { 'agent_id': 'agent_0',
(pid=24022)                            'env_id': 7,
(pid=24022)                            'info': {},
(pid=24022)                            'obs': np.ndarray((2,), dtype=float64,
min=1.011, max=1.011, mean=1.011),
(pid=24022)                            'prev_action': np.ndarray((), dtype=int64,
```

```
min=0.0, max=0.0, mean=0.0),
(pid=24022)                                'prev_reward': 0.0,
(pid=24022)                                'rnn_state': []},
(pid=24022)                    'type': 'PolicyEvalData'}],
(pid=24022)    'agent_1': [ { 'data': { 'agent_id': 'agent_1',
(pid=24022)                             'env_id': 0,
(pid=24022)                             'info': {},
(pid=24022)                             'obs': np.ndarray((2,), dtype=float64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'prev_reward': 0.0,
(pid=24022)                             'rnn_state': []},
(pid=24022)                    'type': 'PolicyEvalData'},
(pid=24022)                  { 'data': { 'agent_id': 'agent_1',
(pid=24022)                             'env_id': 1,
(pid=24022)                             'info': {},
(pid=24022)                             'obs': np.ndarray((2,), dtype=float64,
min=0.707, max=0.707, mean=0.707),
(pid=24022)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'prev_reward': 0.0,
(pid=24022)                             'rnn_state': []},
(pid=24022)                    'type': 'PolicyEvalData'},
(pid=24022)                  { 'data': { 'agent_id': 'agent_1',
(pid=24022)                             'env_id': 2,
(pid=24022)                             'info': {},
(pid=24022)                             'obs': np.ndarray((2,), dtype=float64,
min=0.666, max=0.666, mean=0.666),
(pid=24022)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'prev_reward': 0.0,
(pid=24022)                             'rnn_state': []},
(pid=24022)                    'type': 'PolicyEvalData'},
(pid=24022)                  { 'data': { 'agent_id': 'agent_1',
(pid=24022)                             'env_id': 3,
(pid=24022)                             'info': {},
(pid=24022)                             'obs': np.ndarray((2,), dtype=float64,
min=-0.521, max=-0.521, mean=-0.521),
(pid=24022)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'prev_reward': 0.0,
(pid=24022)                             'rnn_state': []},
(pid=24022)                    'type': 'PolicyEvalData'},
(pid=24022)                  { 'data': { 'agent_id': 'agent_1',
(pid=24022)                             'env_id': 4,
(pid=24022)                             'info': {},
(pid=24022)                             'obs': np.ndarray((2,), dtype=float64,
min=0.631, max=0.631, mean=0.631),
(pid=24022)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'prev_reward': 0.0,
(pid=24022)                             'rnn_state': []},
(pid=24022)                    'type': 'PolicyEvalData'},
(pid=24022)                  { 'data': { 'agent_id': 'agent_1',
(pid=24022)                             'env_id': 5,
(pid=24022)                             'info': {},
(pid=24022)                             'obs': np.ndarray((2,), dtype=float64,
min=-0.45, max=-0.45, mean=-0.45),
```

```
(pid=24022)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                               'prev_reward': 0.0,
(pid=24022)                               'rnn_state': []},
(pid=24022)                     'type': 'PolicyEvalData'},
(pid=24022)                   { 'data': { 'agent_id': 'agent_1',
(pid=24022)                               'env_id': 6,
(pid=24022)                               'info': {},
(pid=24022)                               'obs': np.ndarray((2,), dtype=float64,
min=-0.416, max=-0.416, mean=-0.416),
(pid=24022)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                               'prev_reward': 0.0,
(pid=24022)                               'rnn_state': []},
(pid=24022)                     'type': 'PolicyEvalData'},
(pid=24022)                   { 'data': { 'agent_id': 'agent_1',
(pid=24022)                               'env_id': 7,
(pid=24022)                               'info': {},
(pid=24022)                               'obs': np.ndarray((2,), dtype=float64,
min=1.011, max=1.011, mean=1.011),
(pid=24022)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                               'prev_reward': 0.0,
(pid=24022)                               'rnn_state': []},
(pid=24022)                     'type': 'PolicyEvalData'}]}
(pid=24022)
(pid=24022) 2019-10-05 12:10:22,213        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=24022) 2019-10-05 12:10:22,331        INFO sampler.py:548 -- Outputs of
compute_actions():
(pid=24022)
(pid=24022) { 'agent_0': ( np.ndarray((8,), dtype=int64, min=1.0, max=13.0,
mean=7.625),
(pid=24022)                 [],
(pid=24022)                 { 'q_values': np.ndarray((8, 15), dtype=float32,
min=-0.385, max=2.072, mean=0.436)}),
(pid=24022)   'agent_1': ( np.ndarray((8,), dtype=int64, min=1.0, max=14.0,
mean=6.0),
(pid=24022)                 [],
(pid=24022)                 { 'q_values': np.ndarray((8, 15), dtype=float32,
min=-0.275, max=0.989, mean=0.159)})}
(pid=24022)
(pid=24022) 2019-10-05 12:10:22,719        INFO sample_batch_builder.py:161 --
Trajectory fragment after postprocess_trajectory():
(pid=24022)
(pid=24022) { 'agent_0': { 'data': { 'actions': np.ndarray((32,), dtype=int64,
min=0.0, max=14.0, mean=6.938),
(pid=24022)                          'agent_index': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                          'dones': np.ndarray((32,), dtype=bool,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                          'eps_id': np.ndarray((32,), dtype=int64,
min=1464506375.0, max=1464506375.0, mean=1464506375.0),
(pid=24022)                          'infos': np.ndarray((32,), dtype=object,
head={'delta': -0.8245116503502558}),
(pid=24022)                          'new_obs': np.ndarray((32, 2), dtype=float32,
min=-1.666, max=1.76, mean=0.05),
(pid=24022)                          'obs': np.ndarray((32, 2), dtype=float32,
```

```
min=-1.666, max=1.76, mean=0.081),
(pid=24022)                             'prev_actions': np.ndarray((32,),
dtype=int64, min=0.0, max=14.0, mean=6.906),
(pid=24022)                             'prev_rewards': np.ndarray((32,),
dtype=float32, min=0.0, max=0.417, mean=0.276),
(pid=24022)                             'q_values': np.ndarray((32, 15),
dtype=float32, min=-0.54, max=4.938, mean=0.866),
(pid=24022)                             'rewards': np.ndarray((32,), dtype=float32,
min=0.371, max=1.091, mean=0.827),
(pid=24022)                             't': np.ndarray((32,), dtype=int64, min=0.0,
max=31.0, mean=15.5),
(pid=24022)                             'unroll_id': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'weights': np.ndarray((32,), dtype=float32,
min=2.273, max=2.898, mean=2.427)},
(pid=24022)                  'type': 'SampleBatch'},
(pid=24022)   'agent_1': { 'data': { 'actions': np.ndarray((32,), dtype=int64,
min=0.0, max=14.0, mean=7.531),
(pid=24022)                             'agent_index': np.ndarray((32,), dtype=int64,
min=1.0, max=1.0, mean=1.0),
(pid=24022)                             'dones': np.ndarray((32,), dtype=bool,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'eps_id': np.ndarray((32,), dtype=int64,
min=1464506375.0, max=1464506375.0, mean=1464506375.0),
(pid=24022)                             'infos': np.ndarray((32,), dtype=object,
head={'delta': 1.373416199103248}),
(pid=24022)                             'new_obs': np.ndarray((32, 2), dtype=float32,
min=-1.666, max=1.76, mean=0.05),
(pid=24022)                             'obs': np.ndarray((32, 2), dtype=float32,
min=-1.666, max=1.76, mean=0.081),
(pid=24022)                             'prev_actions': np.ndarray((32,),
dtype=int64, min=0.0, max=14.0, mean=7.094),
(pid=24022)                             'prev_rewards': np.ndarray((32,),
dtype=float32, min=0.0, max=0.41, mean=0.274),
(pid=24022)                             'q_values': np.ndarray((32, 15),
dtype=float32, min=-0.977, max=2.063, mean=0.336),
(pid=24022)                             'rewards': np.ndarray((32,), dtype=float32,
min=0.109, max=1.055, mean=0.778),
(pid=24022)                             't': np.ndarray((32,), dtype=int64, min=0.0,
max=31.0, mean=15.5),
(pid=24022)                             'unroll_id': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24022)                             'weights': np.ndarray((32,), dtype=float32,
min=2.287, max=2.635, mean=2.369)},
(pid=24022)                  'type': 'SampleBatch'}}
(pid=24022)
(pid=24022) 2019-10-05 12:10:22,777        INFO rollout_worker.py:485 -- Completed
sample batch:
(pid=24022)
(pid=24022) { 'count': 256,
(pid=24022)   'policy_batches': { 'agent_0': { 'data': { 'actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=7.176),
(pid=24022)                                              'agent_index':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24022)                                              'dones':
np.ndarray((256,), dtype=bool, min=0.0, max=0.0, mean=0.0),
(pid=24022)                                              'eps_id':
np.ndarray((256,), dtype=int64, min=511967761.0, max=1832562928.0,
mean=1117305429.625),
```

```
(pid=24022)                                                 'infos':
np.ndarray((256,), dtype=object, head={'delta': -0.8245116503502558}),
(pid=24022)                                                 'new_obs':
np.ndarray((256, 2), dtype=float32, min=-1.736, max=1.81, mean=0.01),
(pid=24022)                                                 'obs': np.ndarray((256,
2), dtype=float32, min=-1.736, max=1.883, mean=0.046),
(pid=24022)                                                 'prev_actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.992),
(pid=24022)                                                 'prev_rewards':
np.ndarray((256,), dtype=float32, min=0.0, max=0.42, mean=0.266),
(pid=24022)                                                 'q_values':
np.ndarray((256, 15), dtype=float32, min=-0.733, max=5.536, mean=0.992),
(pid=24022)                                                 'rewards':
np.ndarray((256,), dtype=float32, min=0.12, max=1.149, mean=0.779),
(pid=24022)                                                 't': np.ndarray((256,),
dtype=int64, min=0.0, max=31.0, mean=15.5),
(pid=24022)                                                 'unroll_id':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24022)                                                 'weights':
np.ndarray((256,), dtype=float32, min=2.233, max=2.962, mean=2.458)},
(pid=24022)                                       'type': 'SampleBatch'},
(pid=24022)                           'agent_1': { 'data': { 'actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.855),
(pid=24022)                                                 'agent_index':
np.ndarray((256,), dtype=int64, min=1.0, max=1.0, mean=1.0),
(pid=24022)                                                 'dones':
np.ndarray((256,), dtype=bool, min=0.0, max=0.0, mean=0.0),
(pid=24022)                                                 'eps_id':
np.ndarray((256,), dtype=int64, min=511967761.0, max=1832562928.0,
mean=1117305429.625),
(pid=24022)                                                 'infos':
np.ndarray((256,), dtype=object, head={'delta': 1.373416199103248}),
(pid=24022)                                                 'new_obs':
np.ndarray((256, 2), dtype=float32, min=-1.736, max=1.81, mean=0.01),
(pid=24022)                                                 'obs': np.ndarray((256,
2), dtype=float32, min=-1.736, max=1.883, mean=0.046),
(pid=24022)                                                 'prev_actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.652),
(pid=24022)                                                 'prev_rewards':
np.ndarray((256,), dtype=float32, min=0.0, max=0.42, mean=0.276),
(pid=24022)                                                 'q_values':
np.ndarray((256, 15), dtype=float32, min=-1.033, max=2.068, mean=0.375),
(pid=24022)                                                 'rewards':
np.ndarray((256,), dtype=float32, min=0.109, max=1.163, mean=0.803),
(pid=24022)                                                 't': np.ndarray((256,),
dtype=int64, min=0.0, max=31.0, mean=15.5),
(pid=24022)                                                 'unroll_id':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24022)                                                 'weights':
np.ndarray((256,), dtype=float32, min=2.244, max=2.697, mean=2.376)},
(pid=24022)                                       'type': 'SampleBatch'}},
(pid=24022)   'type': 'MultiAgentBatch'}
(pid=24022)
(pid=24034) /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=24034)   _np_qint8 = np.dtype([("qint8", np.int8, 1)])
(pid=24034) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
```

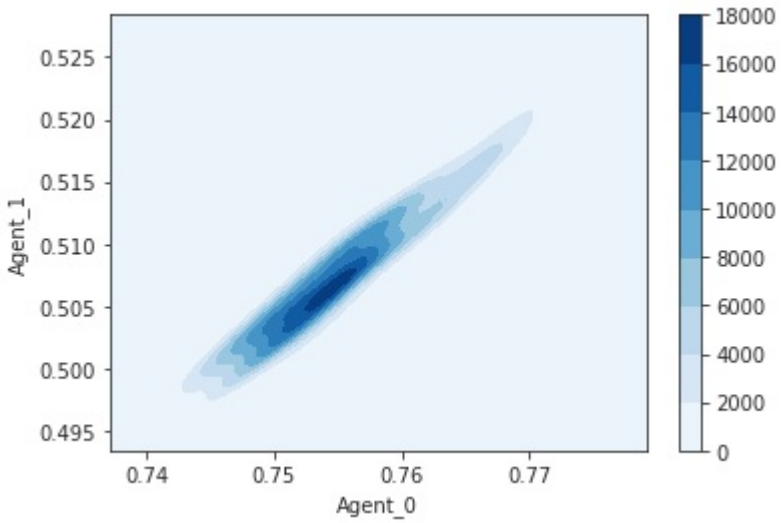
```
tensorflow/python/framework/dtypes.py:517: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24034)   _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
(pid=24034) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
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=24034)   _np_qint16 = np.dtype([("qint16", np.int16, 1)])
(pid=24034) /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=24034)   _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
(pid=24034) /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=24034)   _np_qint32 = np.dtype([("qint32", np.int32, 1)])
(pid=24034) /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=24034)   np_resource = np.dtype([("resource", np.ubyte, 1)])
(pid=24034) /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=24034)   _np_qint8 = np.dtype([("qint8", np.int8, 1)])
(pid=24034) /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=24034)   _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
(pid=24034) /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=24034)   _np_qint16 = np.dtype([("qint16", np.int16, 1)])
(pid=24034) /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=24034)   _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
(pid=24034) /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=24034)   _np_qint32 = np.dtype([("qint32", np.int32, 1)])
(pid=24034) /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=24034)   np_resource = np.dtype([("resource", np.ubyte, 1)])
(pid=24034) 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=24034) Instructions for updating:
```

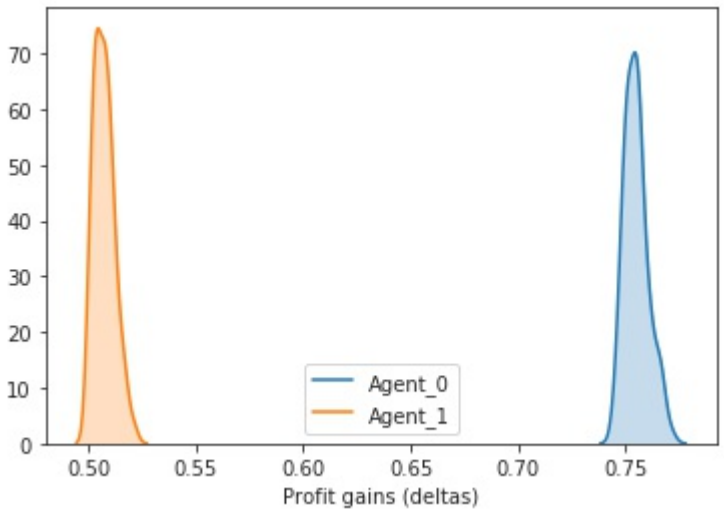
```
(pid=24034) non-resource variables are not supported in the long term
Episode reward 5924.488111362871
Episode 1 of 100
Episode reward 5919.457675778948
Episode 2 of 100
Episode reward 5937.670348793445
Episode 3 of 100
Episode reward 5942.407256528537
Episode 4 of 100
Episode reward 5937.3515609209935
Episode 5 of 100
Episode reward 5914.0388450362725
Episode 6 of 100
Episode reward 5918.70613330432
Episode 7 of 100
Episode reward 5925.513665078765
Episode 8 of 100
Episode reward 5928.328822347925
Episode 9 of 100
Episode reward 5950.9803088369945
Episode 10 of 100
Episode reward 5942.883609241205
Episode 11 of 100
Episode reward 5928.044417218142
Episode 12 of 100
Episode reward 5915.4073898998895
Episode 13 of 100
Episode reward 5917.131223221769
Episode 14 of 100
Episode reward 5907.301464567373
Episode 15 of 100
Episode reward 5935.091569717672
Episode 16 of 100
Episode reward 5950.715079410753
Episode 17 of 100
Episode reward 5921.525215406513
Episode 18 of 100
Episode reward 5915.795994716688
Episode 19 of 100
Episode reward 5916.764841580644
Episode 20 of 100
Episode reward 5912.281106468846
Episode 21 of 100
Episode reward 5925.401629030471
Episode 22 of 100
Episode reward 5919.655458081089
Episode 23 of 100
Episode reward 5937.718749987649
Episode 24 of 100
Episode reward 5923.561231714095
Episode 25 of 100
Episode reward 5921.742416093815
Episode 26 of 100
Episode reward 5950.454191993199
Episode 27 of 100
Episode reward 5913.036035596171
Episode 28 of 100
Episode reward 5930.503828380524
Episode 29 of 100
```

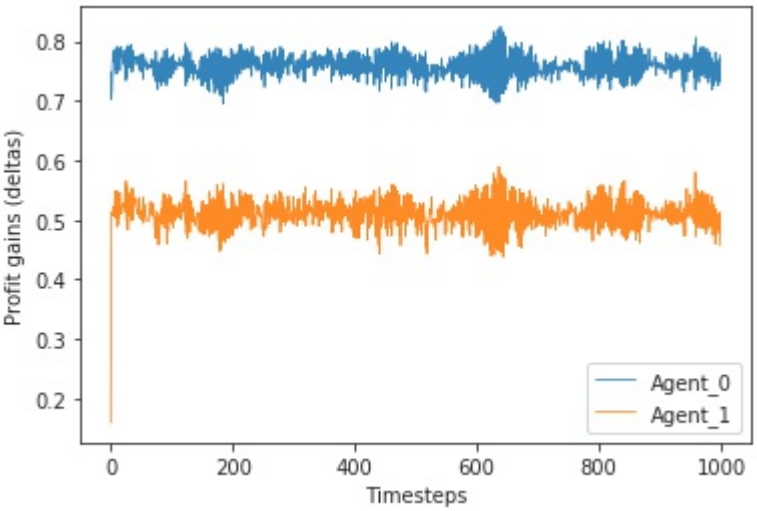
```
Episode reward 5932.989224852297
Episode 30 of 100
Episode reward 5933.840563358838
Episode 31 of 100
Episode reward 5943.502741448723
Episode 32 of 100
Episode reward 5917.4393599493205
Episode 33 of 100
Episode reward 5917.236983351584
Episode 34 of 100
Episode reward 5960.7345108199725
Episode 35 of 100
Episode reward 5914.012380607195
Episode 36 of 100
Episode reward 5928.815563675118
Episode 37 of 100
Episode reward 5937.411558405582
Episode 38 of 100
Episode reward 5914.0645062823
Episode 39 of 100
Episode reward 5928.451763579939
Episode 40 of 100
Episode reward 5934.185856267655
Episode 41 of 100
Episode reward 5911.721143595082
Episode 42 of 100
Episode reward 5920.8582983793885
Episode 43 of 100
Episode reward 5914.190749263307
Episode 44 of 100
Episode reward 5914.198229432338
Episode 45 of 100
Episode reward 5945.917871265167
Episode 46 of 100
Episode reward 5931.375638165351
Episode 47 of 100
Episode reward 5907.898958100883
Episode 48 of 100
Episode reward 5947.5359925429075
Episode 49 of 100
Episode reward 5921.192198725196
Episode 50 of 100
Episode reward 5930.6062745172885
Episode 51 of 100
Episode reward 5926.842875224034
Episode 52 of 100
Episode reward 5923.632112144972
Episode 53 of 100
Episode reward 5930.653172375254
Episode 54 of 100
Episode reward 5917.791845140567
Episode 55 of 100
Episode reward 5929.409949544625
Episode 56 of 100
Episode reward 5916.427004976895
Episode 57 of 100
Episode reward 5932.042352667825
Episode 58 of 100
Episode reward 5918.82452530988
```

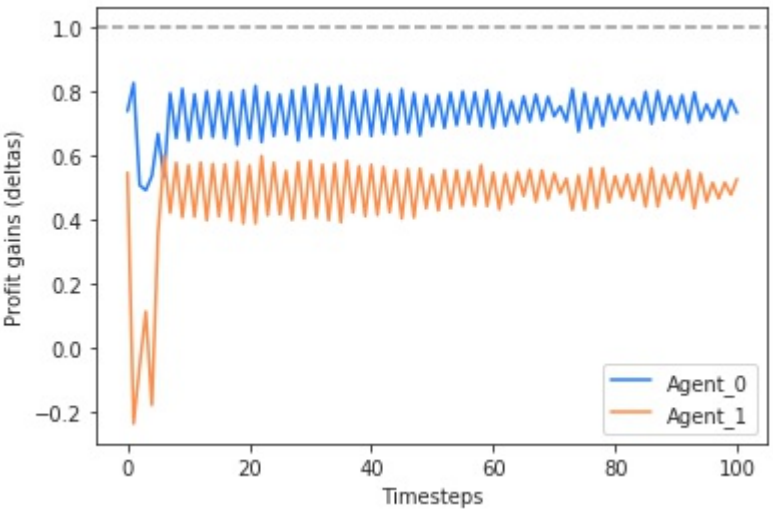
```
Episode 59 of 100
Episode reward 5917.587082688494
Episode 60 of 100
Episode reward 5911.786727414627
Episode 61 of 100
Episode reward 5918.872252933043
Episode 62 of 100
Episode reward 5928.8689069033235
Episode 63 of 100
Episode reward 5919.357947047474
Episode 64 of 100
Episode reward 5928.590669909088
Episode 65 of 100
Episode reward 5915.804597840093
Episode 66 of 100
Episode reward 5910.010836017219
Episode 67 of 100
Episode reward 5928.36926247994
Episode 68 of 100
Episode reward 5921.830484237227
Episode 69 of 100
Episode reward 5919.687178764313
Episode 70 of 100
Episode reward 5931.714916453863
Episode 71 of 100
Episode reward 5941.00842392875
Episode 72 of 100
Episode reward 5922.052911214666
Episode 73 of 100
Episode reward 5939.957648652598
Episode 74 of 100
Episode reward 5940.192424328259
Episode 75 of 100
Episode reward 5932.606197503831
Episode 76 of 100
Episode reward 5927.570601238485
Episode 77 of 100
Episode reward 5910.774331066376
Episode 78 of 100
Episode reward 5920.978647513477
Episode 79 of 100
Episode reward 5936.597322413523
Episode 80 of 100
Episode reward 5926.104805984169
Episode 81 of 100
Episode reward 5928.380701943025
Episode 82 of 100
Episode reward 5916.377818406145
Episode 83 of 100
Episode reward 5924.775869417495
Episode 84 of 100
Episode reward 5948.755692905788
Episode 85 of 100
Episode reward 5931.802112500145
Episode 86 of 100
Episode reward 5926.948136686422
Episode 87 of 100
Episode reward 5921.633501862594
Episode 88 of 100
```

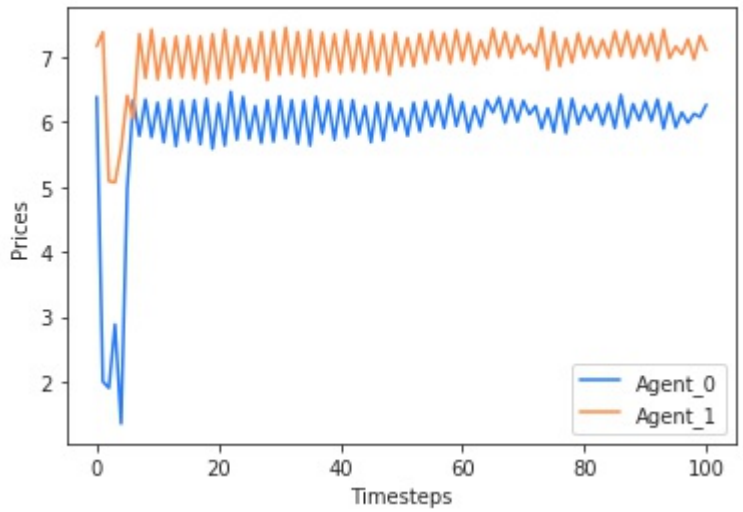
```
Episode reward 5914.078630433585
Episode 89 of 100
Episode reward 5925.623425678302
Episode 90 of 100
Episode reward 5954.453065936898
Episode 91 of 100
Episode reward 5911.785877119069
Episode 92 of 100
Episode reward 5927.434225817144
Episode 93 of 100
Episode reward 5929.4162117672495
Episode 94 of 100
Episode reward 5910.431169466098
Episode 95 of 100
Episode reward 5923.106339018131
Episode 96 of 100
Episode reward 5920.320347573961
Episode 97 of 100
Episode reward 5930.771482724065
Episode 98 of 100
Episode reward 5937.862616150113
Episode 99 of 100
Episode reward 5930.68134033236
Overall deltas mean: 0.6313 and std: 0.1239
Agent0 deltas mean: 0.7551 and std: 0.0055
Agent1 deltas mean: 0.5074 and std: 0.0047
/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)
```




70
60
50
40
30
20
10
0
0.50
0.55
0.60
0.65
0.70
0.75
Profit gains (deltas)
Agent_0
Agent_1


0.8
0.7
0.6
0.5
0.4
0.3
0.2
Profit gains (deltas)
0
200
400
600
800
1000
Timesteps
Agent_0
Agent_1


1.0
0.8
0.6
0.4
0.2
0.0
−0.2
Profit gains (deltas)
0
20
40
60
80
100
Timesteps
Agent_0
Agent_1




```
Traceback (most recent call last):

  File "<ipython-input-1-71c5063fbdbf>", line 1, in <module>
    runfile('/home/lorenzo/algorithmic-pricing/rollout/rollout.py', args='/home/
lorenzo/algorithmic-pricing/train_results/Azure_ApexDQN_Disc/azure_disc_10_res2/
APEX_MultiAgentFirmsPricing_0_2019-09-21_14-03-453aaxi5sb/checkpoint_540/
checkpoint-540 --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]:
```