```
(pid=24289)         'agent_1': np.ndarray((2,), dtype=int64, min=3.0, max=3.0,
mean=3.0)},
(pid=24289)    2: { 'agent_0': np.ndarray((2,), dtype=int64, min=8.0, max=8.0,
mean=8.0),
(pid=24289)         'agent_1': np.ndarray((2,), dtype=int64, min=8.0, max=8.0,
mean=8.0)},
(pid=24289)    3: { 'agent_0': np.ndarray((2,), dtype=int64, min=14.0, max=14.0,
mean=14.0),
(pid=24289)         'agent_1': np.ndarray((2,), dtype=int64, min=14.0, max=14.0,
mean=14.0)},
(pid=24289)    4: { 'agent_0': np.ndarray((2,), dtype=int64, min=7.0, max=7.0,
mean=7.0),
(pid=24289)         'agent_1': np.ndarray((2,), dtype=int64, min=7.0, max=7.0,
mean=7.0)},
(pid=24289)    5: { 'agent_0': np.ndarray((2,), dtype=int64, min=6.0, max=6.0,
mean=6.0),
(pid=24289)         'agent_1': np.ndarray((2,), dtype=int64, min=6.0, max=6.0,
mean=6.0)},
(pid=24289)    6: { 'agent_0': np.ndarray((2,), dtype=int64, min=5.0, max=5.0,
mean=5.0),
(pid=24289)         'agent_1': np.ndarray((2,), dtype=int64, min=5.0, max=5.0,
mean=5.0)},
(pid=24289)    7: { 'agent_0': np.ndarray((2,), dtype=int64, min=3.0, max=3.0,
mean=3.0),
(pid=24289)         'agent_1': np.ndarray((2,), dtype=int64, min=3.0, max=3.0,
mean=3.0)}}
(pid=24289) 2019-10-05 12:46:30,796        INFO sampler.py:305 -- Info return from
env: { 0: {'agent_0': {}, 'agent_1': {}},
(pid=24289)    1: {'agent_0': {}, 'agent_1': {}},
(pid=24289)    2: {'agent_0': {}, 'agent_1': {}},
(pid=24289)    3: {'agent_0': {}, 'agent_1': {}},
(pid=24289)    4: {'agent_0': {}, 'agent_1': {}},
(pid=24289)    5: {'agent_0': {}, 'agent_1': {}},
(pid=24289)    6: {'agent_0': {}, 'agent_1': {}},
(pid=24289)    7: {'agent_0': {}, 'agent_1': {}}}
(pid=24289) 2019-10-05 12:46:30,796        INFO sampler.py:403 -- Preprocessed
obs: np.ndarray((2,), dtype=int64, min=4.0, max=4.0, mean=4.0)
(pid=24289) 2019-10-05 12:46:30,797        INFO sampler.py:407 -- Filtered obs:
np.ndarray((2,), dtype=float64, min=0.0, max=0.0, mean=0.0)
(pid=24289) 2019-10-05 12:46:30,806        INFO sampler.py:521 -- Inputs to
compute_actions():
(pid=24289)
(pid=24289) { 'agent_0': [ { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 0,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                            'prev_reward': 0.0,
(pid=24289)                            'rnn_state': []},
(pid=24289)                  'type': 'PolicyEvalData'},
(pid=24289)                { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 1,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.707, max=-0.707, mean=-0.707),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
```

```
(pid=24289)                                'prev_reward': 0.0,
(pid=24289)                                'rnn_state': []},
(pid=24289)                    'type': 'PolicyEvalData'},
(pid=24289)                { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 2,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=1.134, max=1.134, mean=1.134),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                            'prev_reward': 0.0,
(pid=24289)                            'rnn_state': []},
(pid=24289)                    'type': 'PolicyEvalData'},
(pid=24289)                { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 3,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=1.352, max=1.352, mean=1.352),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                            'prev_reward': 0.0,
(pid=24289)                            'rnn_state': []},
(pid=24289)                    'type': 'PolicyEvalData'},
(pid=24289)                { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 4,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.046, max=-0.046, mean=-0.046),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                            'prev_reward': 0.0,
(pid=24289)                            'rnn_state': []},
(pid=24289)                    'type': 'PolicyEvalData'},
(pid=24289)                { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 5,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.256, max=-0.256, mean=-0.256),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                            'prev_reward': 0.0,
(pid=24289)                            'rnn_state': []},
(pid=24289)                    'type': 'PolicyEvalData'},
(pid=24289)                { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 6,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.471, max=-0.471, mean=-0.471),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                            'prev_reward': 0.0,
(pid=24289)                            'rnn_state': []},
(pid=24289)                    'type': 'PolicyEvalData'},
(pid=24289)                { 'data': { 'agent_id': 'agent_0',
(pid=24289)                            'env_id': 7,
(pid=24289)                            'info': {},
(pid=24289)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.899, max=-0.899, mean=-0.899),
(pid=24289)                            'prev_action': np.ndarray((), dtype=int64,
```

```
min=0.0, max=0.0, mean=0.0),
(pid=24289)                                 'prev_reward': 0.0,
(pid=24289)                                 'rnn_state': []},
(pid=24289)                       'type': 'PolicyEvalData'}],
(pid=24289)    'agent_1': [ { 'data': { 'agent_id': 'agent_1',
(pid=24289)                             'env_id': 0,
(pid=24289)                             'info': {},
(pid=24289)                             'obs': np.ndarray((2,), dtype=float64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'prev_reward': 0.0,
(pid=24289)                             'rnn_state': []},
(pid=24289)                       'type': 'PolicyEvalData'},
(pid=24289)                     { 'data': { 'agent_id': 'agent_1',
(pid=24289)                             'env_id': 1,
(pid=24289)                             'info': {},
(pid=24289)                             'obs': np.ndarray((2,), dtype=float64,
min=-0.707, max=-0.707, mean=-0.707),
(pid=24289)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'prev_reward': 0.0,
(pid=24289)                             'rnn_state': []},
(pid=24289)                       'type': 'PolicyEvalData'},
(pid=24289)                     { 'data': { 'agent_id': 'agent_1',
(pid=24289)                             'env_id': 2,
(pid=24289)                             'info': {},
(pid=24289)                             'obs': np.ndarray((2,), dtype=float64,
min=1.134, max=1.134, mean=1.134),
(pid=24289)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'prev_reward': 0.0,
(pid=24289)                             'rnn_state': []},
(pid=24289)                       'type': 'PolicyEvalData'},
(pid=24289)                     { 'data': { 'agent_id': 'agent_1',
(pid=24289)                             'env_id': 3,
(pid=24289)                             'info': {},
(pid=24289)                             'obs': np.ndarray((2,), dtype=float64,
min=1.352, max=1.352, mean=1.352),
(pid=24289)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'prev_reward': 0.0,
(pid=24289)                             'rnn_state': []},
(pid=24289)                       'type': 'PolicyEvalData'},
(pid=24289)                     { 'data': { 'agent_id': 'agent_1',
(pid=24289)                             'env_id': 4,
(pid=24289)                             'info': {},
(pid=24289)                             'obs': np.ndarray((2,), dtype=float64,
min=-0.046, max=-0.046, mean=-0.046),
(pid=24289)                             'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'prev_reward': 0.0,
(pid=24289)                             'rnn_state': []},
(pid=24289)                       'type': 'PolicyEvalData'},
(pid=24289)                     { 'data': { 'agent_id': 'agent_1',
(pid=24289)                             'env_id': 5,
(pid=24289)                             'info': {},
(pid=24289)                             'obs': np.ndarray((2,), dtype=float64,
min=-0.256, max=-0.256, mean=-0.256),
```

```
(pid=24289)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                               'prev_reward': 0.0,
(pid=24289)                               'rnn_state': []},
(pid=24289)                     'type': 'PolicyEvalData'},
(pid=24289)                   { 'data': { 'agent_id': 'agent_1',
(pid=24289)                               'env_id': 6,
(pid=24289)                               'info': {},
(pid=24289)                               'obs': np.ndarray((2,), dtype=float64,
min=-0.471, max=-0.471, mean=-0.471),
(pid=24289)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                               'prev_reward': 0.0,
(pid=24289)                               'rnn_state': []},
(pid=24289)                     'type': 'PolicyEvalData'},
(pid=24289)                   { 'data': { 'agent_id': 'agent_1',
(pid=24289)                               'env_id': 7,
(pid=24289)                               'info': {},
(pid=24289)                               'obs': np.ndarray((2,), dtype=float64,
min=-0.899, max=-0.899, mean=-0.899),
(pid=24289)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                               'prev_reward': 0.0,
(pid=24289)                               'rnn_state': []},
(pid=24289)                     'type': 'PolicyEvalData'}]}
(pid=24289)
(pid=24289) 2019-10-05 12:46:30,806        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=24289) 2019-10-05 12:46:30,925        INFO sampler.py:548 -- Outputs of
compute_actions():
(pid=24289)
(pid=24289) { 'agent_0': ( np.ndarray((8,), dtype=int64, min=2.0, max=13.0,
mean=5.625),
(pid=24289)                []),
(pid=24289)                { 'q_values': np.ndarray((8, 15), dtype=float32,
min=-1.679, max=1.704, mean=-0.176)}),
(pid=24289)   'agent_1': ( np.ndarray((8,), dtype=int64, min=2.0, max=14.0,
mean=8.125),
(pid=24289)                []),
(pid=24289)                { 'q_values': np.ndarray((8, 15), dtype=float32,
min=-0.977, max=1.159, mean=-0.046)})}
(pid=24289)
(pid=24289) 2019-10-05 12:46:31,314        INFO sample_batch_builder.py:161 --
Trajectory fragment after postprocess_trajectory():
(pid=24289)
(pid=24289) { 'agent_0': { 'data': { 'actions': np.ndarray((32,), dtype=int64,
min=0.0, max=14.0, mean=7.438),
(pid=24289)                          'agent_index': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                          'dones': np.ndarray((32,), dtype=bool,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                          'eps_id': np.ndarray((32,), dtype=int64,
min=1590103218.0, max=1590103218.0, mean=1590103218.0),
(pid=24289)                          'infos': np.ndarray((32,), dtype=object,
head={'delta': 0.9649586297641848}),
(pid=24289)                          'new_obs': np.ndarray((32, 2), dtype=float32,
min=-1.702, max=1.738, mean=-0.005),
(pid=24289)                          'obs': np.ndarray((32, 2), dtype=float32,
```

```
min=-1.702, max=1.738, mean=0.057),
(pid=24289)                             'prev_actions': np.ndarray((32,),
dtype=int64, min=0.0, max=14.0, mean=7.375),
(pid=24289)                             'prev_rewards': np.ndarray((32,),
dtype=float32, min=0.0, max=0.417, mean=0.263),
(pid=24289)                             'q_values': np.ndarray((32, 15),
dtype=float32, min=-3.184, max=2.195, mean=0.099),
(pid=24289)                             'rewards': np.ndarray((32,), dtype=float32,
min=0.327, max=1.079, mean=0.769),
(pid=24289)                             't': np.ndarray((32,), dtype=int64, min=0.0,
max=31.0, mean=15.5),
(pid=24289)                             'unroll_id': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'weights': np.ndarray((32,), dtype=float32,
min=2.26, max=2.55, mean=2.396)},
(pid=24289)                  'type': 'SampleBatch'},
(pid=24289)   'agent_1': { 'data': { 'actions': np.ndarray((32,), dtype=int64,
min=0.0, max=14.0, mean=6.656),
(pid=24289)                             'agent_index': np.ndarray((32,), dtype=int64,
min=1.0, max=1.0, mean=1.0),
(pid=24289)                             'dones': np.ndarray((32,), dtype=bool,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'eps_id': np.ndarray((32,), dtype=int64,
min=1590103218.0, max=1590103218.0, mean=1590103218.0),
(pid=24289)                             'infos': np.ndarray((32,), dtype=object,
head={'delta': 0.049136496697526726}),
(pid=24289)                             'new_obs': np.ndarray((32, 2), dtype=float32,
min=-1.702, max=1.738, mean=-0.005),
(pid=24289)                             'obs': np.ndarray((32, 2), dtype=float32,
min=-1.702, max=1.738, mean=0.057),
(pid=24289)                             'prev_actions': np.ndarray((32,),
dtype=int64, min=0.0, max=14.0, mean=6.406),
(pid=24289)                             'prev_rewards': np.ndarray((32,),
dtype=float32, min=0.0, max=0.417, mean=0.283),
(pid=24289)                             'q_values': np.ndarray((32, 15),
dtype=float32, min=-1.822, max=1.409, mean=-0.106),
(pid=24289)                             'rewards': np.ndarray((32,), dtype=float32,
min=0.193, max=1.068, mean=0.821),
(pid=24289)                             't': np.ndarray((32,), dtype=int64, min=0.0,
max=31.0, mean=15.5),
(pid=24289)                             'unroll_id': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24289)                             'weights': np.ndarray((32,), dtype=float32,
min=2.263, max=2.566, mean=2.385)},
(pid=24289)                  'type': 'SampleBatch'}}
(pid=24289)
(pid=24289) 2019-10-05 12:46:31,369        INFO rollout_worker.py:485 -- Completed
sample batch:
(pid=24289)
(pid=24289) { 'count': 256,
(pid=24289)   'policy_batches': { 'agent_0': { 'data': { 'actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.953),
(pid=24289)                                              'agent_index':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24289)                                              'dones':
np.ndarray((256,), dtype=bool, min=0.0, max=0.0, mean=0.0),
(pid=24289)                                              'eps_id':
np.ndarray((256,), dtype=int64, min=571473033.0, max=1759529298.0,
mean=1220871858.0),
```

```
(pid=24289)                                                 'infos':
np.ndarray((256,), dtype=object, head={'delta': 0.9649586297641848}),
(pid=24289)                                                 'new_obs':
np.ndarray((256, 2), dtype=float32, min=-1.794, max=1.858, mean=-0.032),
(pid=24289)                                                 'obs': np.ndarray((256,
2), dtype=float32, min=-1.794, max=1.858, mean=0.028),
(pid=24289)                                                 'prev_actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.789),
(pid=24289)                                                 'prev_rewards':
np.ndarray((256,), dtype=float32, min=0.0, max=0.42, mean=0.272),
(pid=24289)                                                 'q_values':
np.ndarray((256, 15), dtype=float32, min=-3.579, max=2.245, mean=-0.116),
(pid=24289)                                                 'rewards':
np.ndarray((256,), dtype=float32, min=0.168, max=1.167, mean=0.793),
(pid=24289)                                                 't': np.ndarray((256,),
dtype=int64, min=0.0, max=31.0, mean=15.5),
(pid=24289)                                                 'unroll_id':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24289)                                                 'weights':
np.ndarray((256,), dtype=float32, min=2.187, max=2.851, mean=2.409)},
(pid=24289)                                       'type': 'SampleBatch'},
(pid=24289)                           'agent_1': { 'data': { 'actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.938),
(pid=24289)                                                 'agent_index':
np.ndarray((256,), dtype=int64, min=1.0, max=1.0, mean=1.0),
(pid=24289)                                                 'dones':
np.ndarray((256,), dtype=bool, min=0.0, max=0.0, mean=0.0),
(pid=24289)                                                 'eps_id':
np.ndarray((256,), dtype=int64, min=571473033.0, max=1759529298.0,
mean=1220871858.0),
(pid=24289)                                                 'infos':
np.ndarray((256,), dtype=object, head={'delta': 0.049136496697526726}),
(pid=24289)                                                 'new_obs':
np.ndarray((256, 2), dtype=float32, min=-1.794, max=1.858, mean=-0.032),
(pid=24289)                                                 'obs': np.ndarray((256,
2), dtype=float32, min=-1.794, max=1.858, mean=0.028),
(pid=24289)                                                 'prev_actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.816),
(pid=24289)                                                 'prev_rewards':
np.ndarray((256,), dtype=float32, min=0.0, max=0.42, mean=0.27),
(pid=24289)                                                 'q_values':
np.ndarray((256, 15), dtype=float32, min=-1.946, max=1.731, mean=-0.085),
(pid=24289)                                                 'rewards':
np.ndarray((256,), dtype=float32, min=0.193, max=1.144, mean=0.792),
(pid=24289)                                                 't': np.ndarray((256,),
dtype=int64, min=0.0, max=31.0, mean=15.5),
(pid=24289)                                                 'unroll_id':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24289)                                                 'weights':
np.ndarray((256,), dtype=float32, min=2.226, max=2.662, mean=2.376)},
(pid=24289)                                       'type': 'SampleBatch'}},
(pid=24289)   'type': 'MultiAgentBatch'}
(pid=24289)
(pid=24301) /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=24301)   _np_qint8 = np.dtype([("qint8", np.int8, 1)])
(pid=24301) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
```

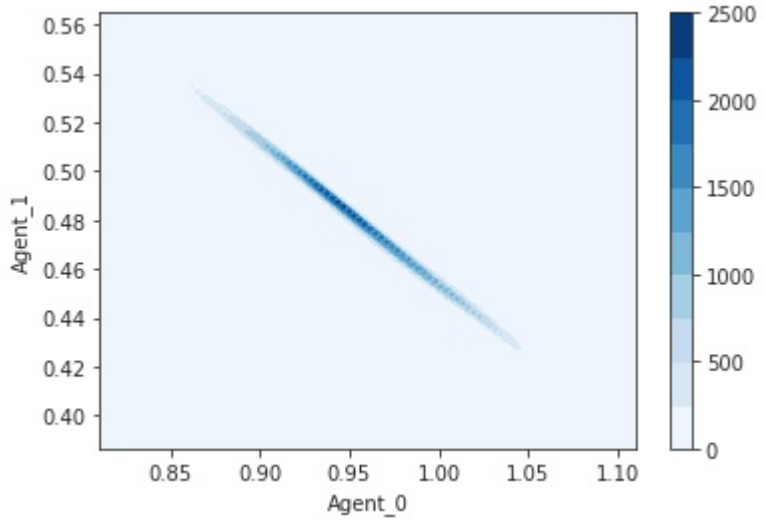
```
tensorflow/python/framework/dtypes.py:517: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24301)   _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
(pid=24301) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
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=24301)   _np_qint16 = np.dtype([("qint16", np.int16, 1)])
(pid=24301) /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=24301)   _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
(pid=24301) /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=24301)   _np_qint32 = np.dtype([("qint32", np.int32, 1)])
(pid=24301) /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=24301)   np_resource = np.dtype([("resource", np.ubyte, 1)])
(pid=24301) /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=24301)   _np_qint8 = np.dtype([("qint8", np.int8, 1)])
(pid=24301) /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=24301)   _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
(pid=24301) /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=24301)   _np_qint16 = np.dtype([("qint16", np.int16, 1)])
(pid=24301) /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=24301)   _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
(pid=24301) /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=24301)   _np_qint32 = np.dtype([("qint32", np.int32, 1)])
(pid=24301) /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=24301)   np_resource = np.dtype([("resource", np.ubyte, 1)])
(pid=24301) 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=24301) Instructions for updating:
```

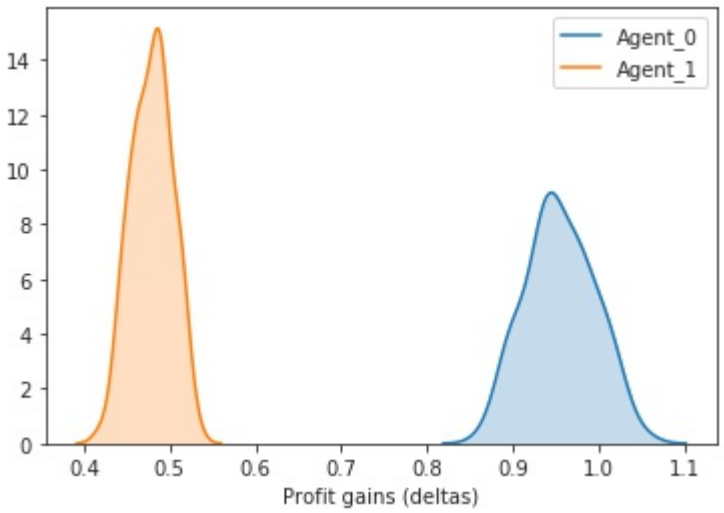
```
(pid=24301) non-resource variables are not supported in the long term
Episode reward 6142.999818321387
Episode 1 of 100
Episode reward 6098.008836567208
Episode 2 of 100
Episode reward 6124.769268310217
Episode 3 of 100
Episode reward 6104.433420555215
Episode 4 of 100
Episode reward 6093.809938748327
Episode 5 of 100
Episode reward 6130.85348871839
Episode 6 of 100
Episode reward 6128.433202183366
Episode 7 of 100
Episode reward 6115.580481173584
Episode 8 of 100
Episode reward 6093.704887124378
Episode 9 of 100
Episode reward 6108.864247875584
Episode 10 of 100
Episode reward 6120.969635273051
Episode 11 of 100
Episode reward 6112.403871562286
Episode 12 of 100
Episode reward 6105.27495729896
Episode 13 of 100
Episode reward 6115.462749354065
Episode 14 of 100
Episode reward 6131.521002792317
Episode 15 of 100
Episode reward 6110.615875293737
Episode 16 of 100
Episode reward 6121.031506578446
Episode 17 of 100
Episode reward 6151.893788714465
Episode 18 of 100
Episode reward 6107.858669230472
Episode 19 of 100
Episode reward 6096.795154760004
Episode 20 of 100
Episode reward 6132.539574166434
Episode 21 of 100
Episode reward 6127.830057595591
Episode 22 of 100
Episode reward 6137.394578644507
Episode 23 of 100
Episode reward 6116.112301843855
Episode 24 of 100
Episode reward 6086.843260043845
Episode 25 of 100
Episode reward 6124.470028658906
Episode 26 of 100
Episode reward 6136.414500899098
Episode 27 of 100
Episode reward 6150.831434562379
Episode 28 of 100
Episode reward 6153.588921373086
Episode 29 of 100
```

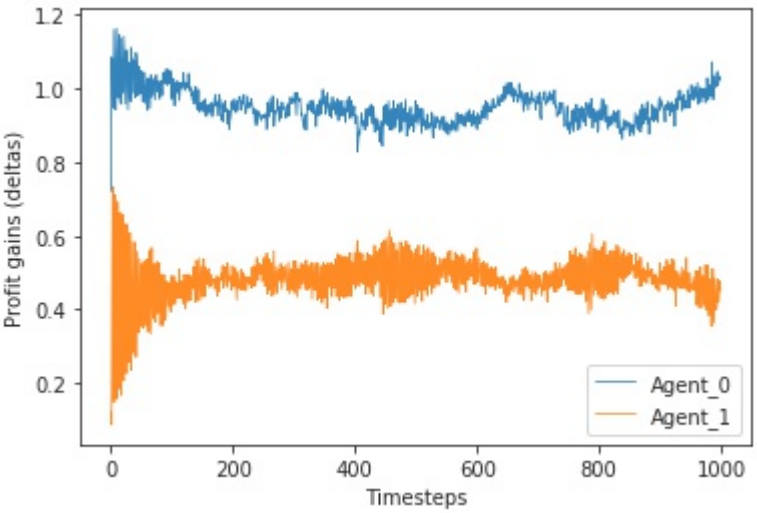
```
Episode reward 6109.625321944033
Episode 30 of 100
Episode reward 6077.84034739002
Episode 31 of 100
Episode reward 6109.98030252912
Episode 32 of 100
Episode reward 6122.289229234435
Episode 33 of 100
Episode reward 6121.891839280291
Episode 34 of 100
Episode reward 6115.787899186581
Episode 35 of 100
Episode reward 6103.131940982645
Episode 36 of 100
Episode reward 6093.512842591608
Episode 37 of 100
Episode reward 6116.725702328719
Episode 38 of 100
Episode reward 6129.293088496286
Episode 39 of 100
Episode reward 6162.520965149427
Episode 40 of 100
Episode reward 6120.890140779872
Episode 41 of 100
Episode reward 6124.516891519952
Episode 42 of 100
Episode reward 6107.033312816347
Episode 43 of 100
Episode reward 6088.934344239701
Episode 44 of 100
Episode reward 6089.829485610938
Episode 45 of 100
Episode reward 6132.243084637938
Episode 46 of 100
Episode reward 6091.837894915975
Episode 47 of 100
Episode reward 6144.762098944912
Episode 48 of 100
Episode reward 6127.500714882022
Episode 49 of 100
Episode reward 6124.5767483408235
Episode 50 of 100
Episode reward 6156.473661157656
Episode 51 of 100
Episode reward 6126.440787931884
Episode 52 of 100
Episode reward 6098.114209720472
Episode 53 of 100
Episode reward 6092.596336415241
Episode 54 of 100
Episode reward 6120.908602524804
Episode 55 of 100
Episode reward 6132.077889688559
Episode 56 of 100
Episode reward 6105.157847590217
Episode 57 of 100
Episode reward 6089.626885134179
Episode 58 of 100
Episode reward 6119.172442303381
```

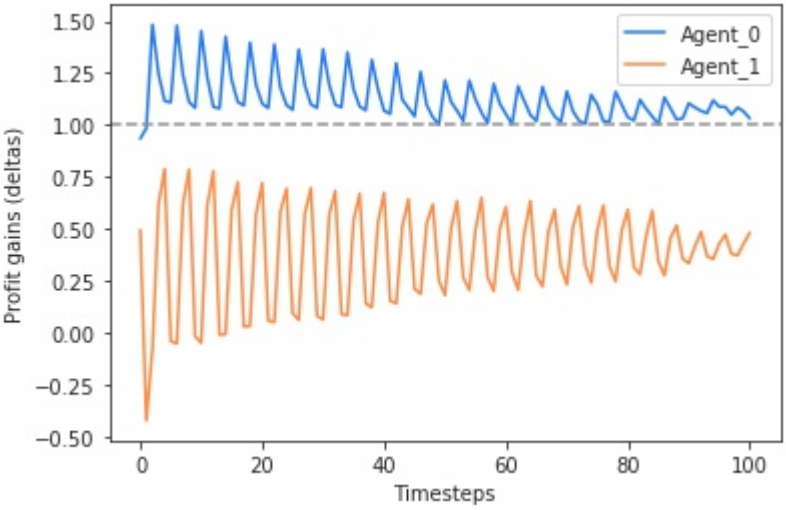
```
Episode 59 of 100
Episode reward 6135.075314596827
Episode 60 of 100
Episode reward 6112.110818490991
Episode 61 of 100
Episode reward 6085.586401588454
Episode 62 of 100
Episode reward 6150.339870046935
Episode 63 of 100
Episode reward 6130.881337132993
Episode 64 of 100
Episode reward 6110.356717448219
Episode 65 of 100
Episode reward 6135.777854287862
Episode 66 of 100
Episode reward 6142.348983742891
Episode 67 of 100
Episode reward 6113.6577024173685
Episode 68 of 100
Episode reward 6116.109884140736
Episode 69 of 100
Episode reward 6108.541759629112
Episode 70 of 100
Episode reward 6109.432432669036
Episode 71 of 100
Episode reward 6130.897638358055
Episode 72 of 100
Episode reward 6085.571517170484
Episode 73 of 100
Episode reward 6147.811724953631
Episode 74 of 100
Episode reward 6113.735366142711
Episode 75 of 100
Episode reward 6145.188632324251
Episode 76 of 100
Episode reward 6139.2626678745855
Episode 77 of 100
Episode reward 6147.694640945014
Episode 78 of 100
Episode reward 6109.511782475405
Episode 79 of 100
Episode reward 6127.51277475987
Episode 80 of 100
Episode reward 6125.781273010686
Episode 81 of 100
Episode reward 6112.9895233148045
Episode 82 of 100
Episode reward 6119.700185267537
Episode 83 of 100
Episode reward 6121.779179835223
Episode 84 of 100
Episode reward 6144.371600396169
Episode 85 of 100
Episode reward 6096.362588500913
Episode 86 of 100
Episode reward 6118.018436267338
Episode 87 of 100
Episode reward 6109.045801888677
Episode 88 of 100
```

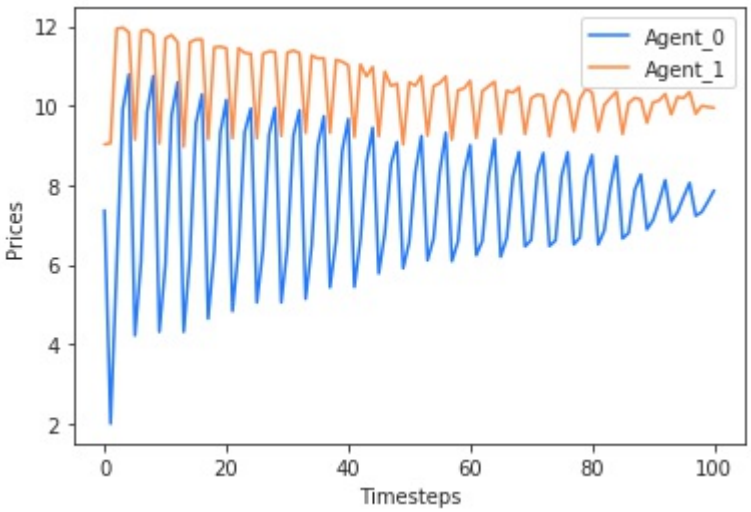
```
Episode reward 6107.196374216542
Episode 89 of 100
Episode reward 6115.616630686939
Episode 90 of 100
Episode reward 6101.040905462786
Episode 91 of 100
Episode reward 6151.455966173686
Episode 92 of 100
Episode reward 6140.461747960009
Episode 93 of 100
Episode reward 6111.529227546569
Episode 94 of 100
Episode reward 6127.041421305856
Episode 95 of 100
Episode reward 6112.510051579756
Episode 96 of 100
Episode reward 6145.014045534442
Episode 97 of 100
Episode reward 6107.670504177145
Episode 98 of 100
Episode reward 6109.174397514898
Episode 99 of 100
Episode reward 6113.173313025098
Overall deltas mean: 0.7176 and std: 0.2405
Agent0 deltas mean: 0.9559 and std: 0.0397
Agent1 deltas mean: 0.4793 and std: 0.0234
/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
14
12
10
8
6
4
2
0
0.4
0.5
0.6
0.7
0.8
0.9
1.0
1.1
Profit gains (deltas)


1.2
1.0
0.8
0.6
0.4
0.2
Profit gains (deltas)
Agent_0
Agent_1
0
200
400
600
800
1000
Timesteps


1.50
1.25
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
Profit gains (deltas)
Agent_0
Agent_1
0
20
40
60
80
100
Timesteps




```
Traceback (most recent call last):

  File "<ipython-input-1-bdf2bab3a9ef>", 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_280/checkpoint-280 --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]:
```