Learning Algorithm

The DDPG algorithm was used on this challenge, modeling a similar approach used in the second project for continuous control. As most of the code worked well, there were aspects (such as gradient clipping) that I used from the last project by default.

I only marginally changed the hyperparameters beyond the ones that worked for the last project, but there were many different variables beyond the hyperparameters (such as the use of noise added to the action space) that I found affected the ability for the agent to learn effectively and within a reasonable period of time. Generally, I do not perceive that the hyperparameters were as important as the architecture of the neural networks or the use of these other variables such as noise (but that will be covered in more detail below). I observed far more variability of training effectiveness when changing the neural network architecture than I did when I made changes to hyperparameter settings, such as changing the size of the replay buffer.

The Actor neural network architecture I used had four hidden layers, with the input and output mapping of: *<state_size(24)> → 256 → 128 → 32 → 8 (not parameterized) → <action_size(2)>*

I experimented with a few different architectures. The four hidden layer architecture outlined above (ironically) worked as well as any other. I did try a wider network (512 → 256 → 64 → 16) in the middle between the observation and action spaces) but it was incredibly slow to execute (back propagation seemed to take forever) and it never even learned anything (tried twice). I also tried smaller hidden layer size parameters but never got effective (much less consistent) learning results.

Even though I did over 20 unique runs (my administration wasn't consistent enough to document all 20, but I tried to capture most!), I achieved good results in only about five runs, achieving the objective values in differing amounts, since I wanted to train (and then test) an agent that achieved a level of proficiency far above 0.5. I never expected to do so many unique runs, and so my documentation of them is inconsistent. Appendix A is what I documented in my notes.

The Critic neural network architecture I used also had four hidden layers. I never seriously investigated having a separate architecture for the Critic network, even though some of the documented results I have observed employ different sizes (in number of hidden layers and neurons per layer). Instead, I used the same basic architecture (same number of hidden layers, with parameterized number of neurons for each) for both Actor and Critic networks, even though the first hidden layer and the output layer necessarily change, given the focus of the Critic network.

As such, the Critic neural network architecture I used had input and output mapping of: *<state_size(24)> → 256 (`cat` to 260) → 128 → 32 → 8 (not parameterized) →* 1*>.*

The hyperparameters used in the implementation were:

```
BUFFER_SIZE = int(1e6)  # replay buffer size
BATCH_SIZE = 128        # minibatch size
GAMMA = 0.99            # discount factor
TAU = 1e-3              # for soft update of target parameters
LR_ACTOR = 1e-4         # learning rate of the actor
LR_CRITIC = 1e-3        # learning rate of the critic
WEIGHT_DECAY = .0001    # L2 weight decay
```

The use of the WEIGHT_DECAY parameter is another area where – like the use of noise – I would like to spend more time in future analysis. The last runs I performed did NOT use it, but I expect it may have allowed results to be more stable.

Results and Plot of Rewards

Per above, training the agent was far more inconsistent (even after stabilizing the code) than I expected. For example, in the latter stages of testing, where I did almost 10 runs in a row with *exactly* the same code and configuration (and even the same initial seed), sometimes the agent would train well, and sometimes it wouldn't (and I would kill the job early to avoid the use of gpu time!). I went back and forth between training in gpu and cpu, because the results were so inconsistent and sometimes a full training run (when trying to achieve a score target of >2.25 for example) would take 5-7 hours (if it learned and unlearned in the process). When it never learned much of anything, the training concluded far faster!

I also ended up getting these (inconsistent, but occasionally strong) results with `add_noise = False`, so at the end of the course (with any remaining gpu time!) I would like to experiment further with `add_noise = True` to see if I can achieve better.

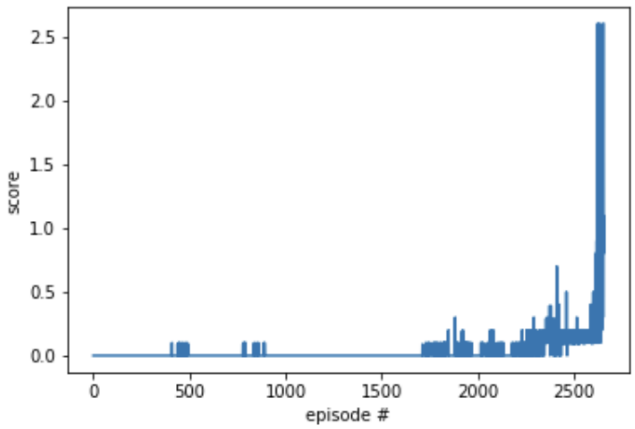
In summary, I did a number of runs with 'solved' = 0.5, and after successfully training a few agents that could deliver on that level of proficiency, increased the target to 1.0, 2.0 and later 2.25.

This is a sample run and performance graph from one of the runs that delivered on proficiency of 0.5.

```
episode [100/3000]      average score: 0.00
episode [200/3000]      average score: 0.00
episode [300/3000]      average score: 0.00
episode [400/3000]      average score: 0.00
episode [500/3000]      average score: 0.02
episode [600/3000]      average score: 0.00
episode [700/3000]      average score: 0.00
episode [800/3000]      average score: 0.01
episode [900/3000]      average score: 0.01
episode [1000/3000]     average score: 0.00
episode [1100/3000]     average score: 0.00
episode [1200/3000]     average score: 0.00
episode [1300/3000]     average score: 0.00
episode [1400/3000]     average score: 0.00
episode [1500/3000]     average score: 0.00
episode [1600/3000]     average score: 0.00
episode [1700/3000]     average score: 0.00
episode [1800/3000]     average score: 0.02
episode [1900/3000]     average score: 0.02
episode [2000/3000]     average score: 0.03
episode [2100/3000]     average score: 0.03
episode [2200/3000]     average score: 0.01
episode [2300/3000]     average score: 0.07
episode [2400/3000]     average score: 0.10
episode [2500/3000]     average score: 0.14
episode [2600/3000]     average score: 0.15

environment solved in 2554 episodes... average score: 0.52

overall average score: 0.04
```



Interestingly, look at the high variance below of the agent on 10 sequential runs. In three of the episodes it successfully made it to the maximum number of time steps (1000), while the minimum was only 1 successful volley over the net. I assume this is very dependent upon the random initialization of the ball:

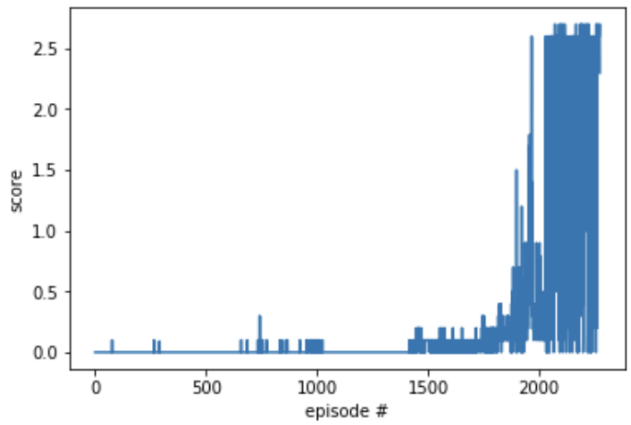
```
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.400   averaged agent score: 0.395     player volleys: 10.8
max agent score: 0.400   averaged agent score: 0.395     player volleys: 12.1
max agent score: 0.200   averaged agent score: 0.145     player volleys: 5.2
max agent score: 1.800   averaged agent score: 1.795     player volleys: 50.4
max agent score: 0.400   averaged agent score: 0.395     player volleys: 12.1
max agent score: 2.600   averaged agent score: 2.550     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.045     player volleys: 1.1
max agent score: 0.500   averaged agent score: 0.495     player volleys: 14.7
max agent score: 2.600   averaged agent score: 2.550     player volleys: 71.5
```

This is a sample run and performance graph from the first run that delivered performance of > 2.0 (notice how, oddly, it did this in less episodes than several of the runs took to achieve proficiency of 0.5).

```
episode [100/5000]      average score: 0.00
episode [200/5000]      average score: 0.00
episode [300/5000]      average score: 0.00
episode [400/5000]      average score: 0.00
episode [500/5000]      average score: 0.00
episode [600/5000]      average score: 0.00
episode [700/5000]      average score: 0.00
episode [800/5000]      average score: 0.01
episode [900/5000]      average score: 0.01
episode [1000/5000]     average score: 0.01
episode [1100/5000]     average score: 0.01
episode [1200/5000]     average score: 0.00
episode [1300/5000]     average score: 0.00
episode [1400/5000]     average score: 0.00
episode [1500/5000]     average score: 0.05
episode [1600/5000]     average score: 0.05
episode [1700/5000]     average score: 0.05
episode [1800/5000]     average score: 0.09
episode [1900/5000]     average score: 0.15
episode [2000/5000]     average score: 0.44
episode [2100/5000]     average score: 1.24
episode [2200/5000]     average score: 1.53

environment solved in 2176 episodes... average score: 2.01

overall average score: 0.23
```



Lastly, I started a notebook later and was able to successfully load the `.pth` files from both Actor and Critic networks (even though I suppose I only need the Actor network) in order to show that the agents could be used to show a highly proficient trained agent. Here, the `*_a` agent in the summary table below was trained with a target of `0.5`, `*_b` was trained with target of `1.0`, and `*_c` was trained with target of `2.0`. The agent trained that improved up to the 2.0 level was able to volley the ball back and forth a full 71+ times (which hit the maximum time steps allowable in the episode of 1000) every single time in this test (but not tests).

```
actor file:  km_actor_a.pth
critic file: km_critic_a.pth
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.045     player volleys: 1.1
max agent score: 0.200   averaged agent score: 0.195     player volleys: 6.1
max agent score: 0.100   averaged agent score: 0.045     player volleys: 2.1
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.095     player volleys: 2.6
max agent score: 0.200   averaged agent score: 0.145     player volleys: 5.1
max agent score: 0.100   averaged agent score: 0.095     player volleys: 4.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.095     player volleys: 2.8
max agent score: 0.200   averaged agent score: 0.145     player volleys: 5.0
max agent score: 0.100   averaged agent score: 0.095     player volleys: 3.9
max agent score: 0.090   averaged agent score: 0.045     player volleys: 2.1
max agent score: 0.100   averaged agent score: 0.095     player volleys: 3.8
max agent score: 0.100   averaged agent score: 0.045     player volleys: 2.1
max agent score: 0.100   averaged agent score: 0.045     player volleys: 2.1
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
average across test run: 0.969   average volleys: 27.2

actor file:  km_actor_b.pth
critic file: km_critic_b.pth
max agent score: 2.490   averaged agent score: 2.445     player volleys: 65.9
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 1.590   averaged agent score: 1.545     player volleys: 41.6
max agent score: 0.600   averaged agent score: 0.495     player volleys: 14.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.200   averaged agent score: 0.145     player volleys: 3.8
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 1.600   averaged agent score: 1.595     player volleys: 43.3
max agent score: 0.600   averaged agent score: 0.545     player volleys: 17.4
max agent score: 0.590   averaged agent score: 0.545     player volleys: 15.4
max agent score: 2.600   averaged agent score: 2.550     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
average across test run: 2.061   average volleys: 56.6

actor file:  km_actor_c.pth
critic file: km_critic_c.pth
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
average across test run: 2.635   average volleys: 71.5
```

Ideas for Future Work

Per above, I was surprised and dismayed at how unpredictable the training process was, even when using all the same code, hyperparameters and even seed values. There was an astounding amount of variation to the results that I can only attribute to the distribution in the initial location and direction of the ball, and the randomness with which early (completely arbitrary) movements happened to correlate with ball locations. Oddly however, it takes a staggering number of data samples in order to start to learn, which seems in conflict with the above statement.

As such, I would be inclined to first investigate how randomness introduced by the `add_noise` parameter either helps or hurts the results. When I first started training (on the single agent version), since the agent wasn't learning effectively (inevitably for other reasons), the `add_noise` flag didn't seem to have a material effect. However, after making major changes (such as neural network architectures), I never went back to considering how `add_noise` would affect the results by turning it off or on: I was able to get decent results within a few runs, so I never tried it again. I would also like to more thoroughly try different learning rate and weight decay values during training, as I wrestled so long with some of the core code that I didn't pursue different permutations extensively.

After building a more robust (and more importantly, predictable) process for solving the problem, I would like to consider the strengths and weaknesses of other algorithms such as PPO, A3C or D4PG in solving the main problem. (I want to read the recommended paper that benchmarks different algorithms for continuous control tasks to consider this before attempting one of them.)

APPENDIX – RUN LOGS

Run1: solved in 1600 episodes (should have done screen capture! I never thought it would solve on the first try)

Run2: fail (zeros entire way)

Run3: solved in 1564 episodes (with architecture in model.py)

```
episode [50/3000]       average score: 0.000 iterations=15
episode [100/3000]      average score: 0.000 iterations=15
episode [150/3000]      average score: 0.001 iterations=14
episode [200/3000]      average score: 0.001 iterations=14
episode [250/3000]      average score: 0.000 iterations=14
episode [300/3000]      average score: 0.000 iterations=14
episode [350/3000]      average score: 0.007 iterations=14
episode [400/3000]      average score: 0.007 iterations=14
episode [450/3000]      average score: 0.000 iterations=14
episode [500/3000]      average score: 0.000 iterations=14
episode [550/3000]      average score: 0.000 iterations=14
episode [600/3000]      average score: 0.000 iterations=15
episode [650/3000]      average score: 0.000 iterations=15
episode [700/3000]      average score: 0.000 iterations=15
episode [750/3000]      average score: 0.000 iterations=15
episode [800/3000]      average score: 0.001 iterations=14
episode [850/3000]      average score: 0.006 iterations=31
episode [900/3000]      average score: 0.016 iterations=15
episode [950/3000]      average score: 0.020 iterations=14
episode [1000/3000]     average score: 0.010 iterations=15
episode [1050/3000]     average score: 0.007 iterations=51
episode [1100/3000]     average score: 0.021 iterations=30
episode [1150/3000]     average score: 0.052 iterations=14
episode [1200/3000]     average score: 0.060 iterations=55
episode [1250/3000]     average score: 0.069 iterations=46
episode [1300/3000]     average score: 0.093 iterations=63
episode [1350/3000]     average score: 0.120 iterations=50
episode [1400/3000]     average score: 0.142 iterations=33
episode [1450/3000]     average score: 0.147 iterations=68
episode [1500/3000]     average score: 0.148 iterations=178
episode [1550/3000]     average score: 0.135 iterations=107
episode [1600/3000]     average score: 0.146 iterations=70
episode [1650/3000]     average score: 0.360 iterations=1001

environment solved in 1564 episodes... average score: 0.506
```

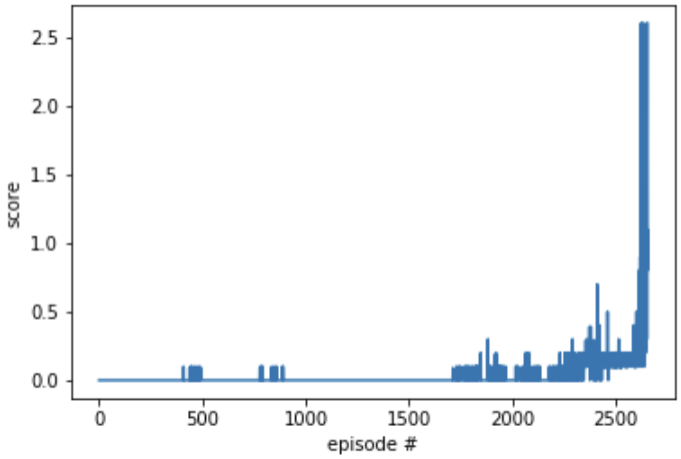
Run4: fail (zeros for 3000 episodes)

Run5: solved in 2554 episodes (target = >0.5)

```
episode [100/3000]      average score: 0.00
episode [200/3000]      average score: 0.00
episode [300/3000]      average score: 0.00
episode [400/3000]      average score: 0.00
episode [500/3000]      average score: 0.02
episode [600/3000]      average score: 0.00
episode [700/3000]      average score: 0.00
episode [800/3000]      average score: 0.01
episode [900/3000]      average score: 0.01
episode [1000/3000]     average score: 0.00
episode [1100/3000]     average score: 0.00
episode [1200/3000]     average score: 0.00
episode [1300/3000]     average score: 0.00
episode [1400/3000]     average score: 0.00
episode [1500/3000]     average score: 0.00
episode [1600/3000]     average score: 0.00
episode [1700/3000]     average score: 0.00
episode [1800/3000]     average score: 0.02
episode [1900/3000]     average score: 0.02
episode [2000/3000]     average score: 0.03
episode [2100/3000]     average score: 0.03
episode [2200/3000]     average score: 0.01
episode [2300/3000]     average score: 0.07
episode [2400/3000]     average score: 0.10
episode [2500/3000]     average score: 0.14
episode [2600/3000]     average score: 0.15

environment solved in 2554 episodes... average score: 0.52

overall average score: 0.04
```




```
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.400   averaged agent score: 0.395     player volleys: 10.8
max agent score: 0.400   averaged agent score: 0.395     player volleys: 12.1
max agent score: 0.200   averaged agent score: 0.145     player volleys: 5.2
max agent score: 1.800   averaged agent score: 1.795     player volleys: 50.4
max agent score: 0.400   averaged agent score: 0.395     player volleys: 12.1
max agent score: 2.600   averaged agent score: 2.550     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.045     player volleys: 1.1
max agent score: 0.500   averaged agent score: 0.495     player volleys: 14.7
max agent score: 2.600   averaged agent score: 2.550     player volleys: 71.5
```

Run6: fail (undulating learning, but never made it)

Run7: set 'solved' to 1.50. Delivered 1.52 in 3213 episodes.

```
episode [100/5000]      average score: 0.00
episode [200/5000]      average score: 0.00
episode [300/5000]      average score: 0.00
episode [400/5000]      average score: 0.01
episode [500/5000]      average score: 0.00
episode [600/5000]      average score: 0.00
episode [700/5000]      average score: 0.00
episode [800/5000]      average score: 0.00
episode [900/5000]      average score: 0.00
episode [1000/5000]     average score: 0.00
episode [1100/5000]     average score: 0.00
episode [1200/5000]     average score: 0.00
episode [1300/5000]     average score: 0.00
episode [1400/5000]     average score: 0.00
episode [1500/5000]     average score: 0.00
episode [1600/5000]     average score: 0.00
episode [1700/5000]     average score: 0.00
episode [1800/5000]     average score: 0.00
episode [1900/5000]     average score: 0.00
episode [2000/5000]     average score: 0.00
episode [2100/5000]     average score: 0.00
episode [2200/5000]     average score: 0.01
episode [2300/5000]     average score: 0.01
episode [2400/5000]     average score: 0.04
episode [2500/5000]     average score: 0.08
episode [2600/5000]     average score: 0.06
episode [2700/5000]     average score: 0.08
episode [2800/5000]     average score: 0.11
episode [2900/5000]     average score: 0.14
episode [3000/5000]     average score: 0.15
episode [3100/5000]     average score: 0.14
episode [3200/5000]     average score: 0.77
episode [3300/5000]     average score: 1.28

environment solved in 3213 episodes... average score: 1.52

overall average score: 0.10
```

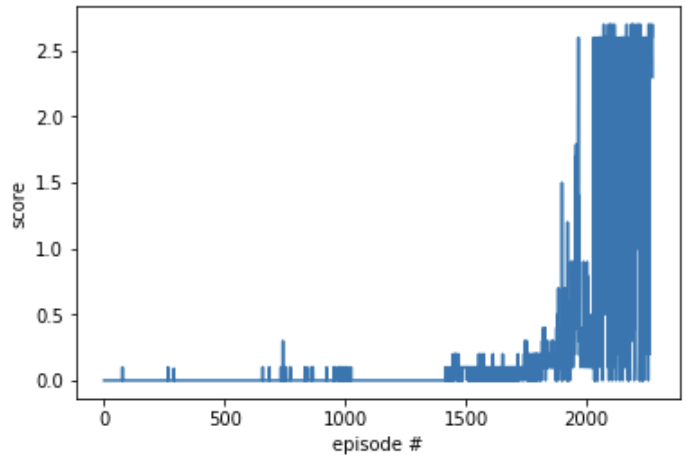
Run8: set 'solved' at 2.0 instead of 1.0, episode target = 10,000. Solved in 2176!! (best run)

```
episode [100/5000]      average score: 0.00
episode [200/5000]      average score: 0.00
episode [300/5000]      average score: 0.00
episode [400/5000]      average score: 0.00
episode [500/5000]      average score: 0.00
episode [600/5000]      average score: 0.00
episode [700/5000]      average score: 0.00
episode [800/5000]      average score: 0.01
episode [900/5000]      average score: 0.01
episode [1000/5000]     average score: 0.01
episode [1100/5000]     average score: 0.01
episode [1200/5000]     average score: 0.00
episode [1300/5000]     average score: 0.00
episode [1400/5000]     average score: 0.00
episode [1500/5000]     average score: 0.05
episode [1600/5000]     average score: 0.05
episode [1700/5000]     average score: 0.05
episode [1800/5000]     average score: 0.09
episode [1900/5000]     average score: 0.15
episode [2000/5000]     average score: 0.44
episode [2100/5000]     average score: 1.24
episode [2200/5000]     average score: 1.53

environment solved in 2176 episodes... average score: 2.01

overall average score: 0.23
```

Learning graph



Validation run with this agent (look at variance! Staggering! Why?)

```
max agent score: 2.550   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.495   averaged agent score: 0.500     player volleys: 14.4
max agent score: 2.450   averaged agent score: 2.500     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.195   averaged agent score: 0.200     player volleys: 5.1
max agent score: -0.005  averaged agent score: 0.000     player volleys: 1.0
max agent score: -0.005  averaged agent score: 0.000     player volleys: 1.0
max agent score: -0.005  averaged agent score: 0.000     player volleys: 1.0
max agent score: 0.045   averaged agent score: 0.100     player volleys: 3.8
max agent score: 0.045   averaged agent score: 0.100     player volleys: 2.6

max agent score: 0.745   averaged agent score: 0.800     player volleys: 24.5
max agent score: 2.245   averaged agent score: 2.300     player volleys: 63.1
max agent score: 2.550   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: -0.005  averaged agent score: 0.000     player volleys: 0.3
max agent score: -0.005  averaged agent score: 0.000     player volleys: 1.0
max agent score: 0.745   averaged agent score: 0.800     player volleys: 24.5
max agent score: 0.045   averaged agent score: 0.100     player volleys: 2.7
max agent score: -0.005  averaged agent score: 0.000     player volleys: 1.0
max agent score: -0.005  averaged agent score: 0.000     player volleys: 1.1

max agent score: 0.145   averaged agent score: 0.200     player volleys: 5.0
max agent score: 0.195   averaged agent score: 0.200     player volleys: 8.3
max agent score: 0.695   averaged agent score: 0.700     player volleys: 20.9
max agent score: -0.005  averaged agent score: 0.000     player volleys: 1.0
max agent score: 0.095   averaged agent score: 0.100     player volleys: 4.9
max agent score: 2.550   averaged agent score: 2.600     player volleys: 71.5
max agent score: 1.745   averaged agent score: 1.800     player volleys: 50.6
max agent score: 2.500   averaged agent score: 2.500     player volleys: 71.5
max agent score: 0.095   averaged agent score: 0.100     player volleys: 3.2
max agent score: 2.550   averaged agent score: 2.600     player volleys: 71.5

max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.700   averaged agent score: 0.695     player volleys: 20.0
max agent score: 0.400   averaged agent score: 0.345     player volleys: 10.6
max agent score: 0.600   averaged agent score: 0.545     player volleys: 19.4
max agent score: 1.300   averaged agent score: 1.245     player volleys: 37.4
max agent score: 0.890   averaged agent score: 0.845     player volleys: 25.8
max agent score: 0.000   averaged agent score: -0.005    player volleys: 1.0
max agent score: 0.000   averaged agent score: -0.005    player volleys: 1.1
max agent score: 0.100   averaged agent score: 0.095     player volleys: 3.8
max agent score: 0.700   averaged agent score: 0.695     player volleys: 21.9
```

Q: what do we see? Max steps = 71.5 volleys (and I set volleys = iterations / ~14, so 1000 must be maximum time steps in the Unity engine, which translates to a possible max score of perfect agent ~2.6 (as observed above). Can we train an agent to get close to that?

Run9: set 'solved' at 2.4. Observed 'undulating' learning, and some dramatic 'unlearning' during certain segments. The job timed out (took 4+ hours), but look at the up-and-down learning.

```
episode [100/5000]      average score: 0.00
episode [200/5000]      average score: 0.00
episode [300/5000]      average score: 0.00
episode [400/5000]      average score: 0.01
episode [500/5000]      average score: 0.03
episode [600/5000]      average score: 0.03
episode [700/5000]      average score: 0.03
episode [800/5000]      average score: 0.02
episode [900/5000]      average score: 0.02
episode [1000/5000]     average score: 0.00
episode [1100/5000]     average score: 0.00
episode [1200/5000]     average score: 0.00
episode [1300/5000]     average score: 0.04
episode [1400/5000]     average score: 0.08
episode [1500/5000]     average score: 0.09
episode [1600/5000]     average score: 0.11
episode [1700/5000]     average score: 0.10
episode [1800/5000]     average score: 0.11
episode [1900/5000]     average score: 0.12
episode [2000/5000]     average score: 0.17
episode [2100/5000]     average score: 0.22
episode [2200/5000]     average score: 1.18
episode [2300/5000]     average score: 0.27
episode [2400/5000]     average score: 0.92
episode [2500/5000]     average score: 0.96
episode [2600/5000]     average score: 0.92
episode [2700/5000]     average score: 0.85
episode [2800/5000]     average score: 0.23
episode [2900/5000]     average score: 0.81
episode [3000/5000]     average score: 0.71
episode [3100/5000]     average score: 0.26
episode [3200/5000]     average score: 0.88
episode [3300/5000]     average score: 1.37
episode [3400/5000]     average score: 1.08
```

Run10: set ‘solved’ at 2.4, episodes = 5000. Never got out of the gates. Zeros the whole way.

```
episode [100/5000]      average score: 0.00
episode [200/5000]      average score: 0.00
episode [300/5000]      average score: 0.00
episode [400/5000]      average score: 0.00
episode [500/5000]      average score: 0.00
episode [600/5000]      average score: 0.00
episode [700/5000]      average score: 0.02
episode [800/5000]      average score: 0.03
episode [900/5000]      average score: 0.00
episode [1000/5000]     average score: 0.00
episode [1100/5000]     average score: 0.00
episode [1200/5000]     average score: 0.00
episode [1300/5000]     average score: 0.00
episode [1400/5000]     average score: 0.00
episode [1500/5000]     average score: 0.00
episode [1600/5000]     average score: 0.00
episode [1700/5000]     average score: 0.00
episode [1800/5000]     average score: 0.00
episode [1900/5000]     average score: 0.00
episode [2000/5000]     average score: 0.00
episode [2100/5000]     average score: 0.00
episode [2200/5000]     average score: 0.00
episode [2300/5000]     average score: 0.00
episode [2400/5000]     average score: 0.00
episode [2500/5000]     average score: 0.00
episode [2600/5000]     average score: 0.00
episode [2700/5000]     average score: 0.00
episode [2800/5000]     average score: 0.00
episode [2900/5000]     average score: 0.00
episode [3000/5000]     average score: 0.00
episode [3100/5000]     average score: 0.00
episode [3200/5000]     average score: 0.00
episode [3300/5000]     average score: 0.00
episode [3400/5000]     average score: 0.00
episode [3500/5000]     average score: 0.00
episode [3600/5000]     average score: 0.00
episode [3700/5000]     average score: 0.00
episode [3800/5000]     average score: 0.00
episode [3900/5000]     average score: 0.00
episode [4000/5000]     average score: 0.00
episode [4100/5000]     average score: 0.00
episode [4200/5000]     average score: 0.00
episode [4300/5000]     average score: 0.00
episode [4400/5000]     average score: 0.00
episode [4500/5000]     average score: 0.00
episode [4600/5000]     average score: 0.00
episode [4700/5000]     average score: 0.00
episode [4800/5000]     average score: 0.00
episode [4900/5000]     average score: 0.00
episode [5000/5000]     average score: 0.00

overall average score: 0.00
```

Run11: changed NN to 3 hidden layers: obs_space → 256 → 64 → 16 → action_space. Epic fail.

```
episode [100/5000]      average score: 0.00
episode [200/5000]      average score: 0.00
episode [300/5000]      average score: 0.00
episode [400/5000]      average score: 0.00
episode [500/5000]      average score: 0.00
episode [600/5000]      average score: 0.00
episode [700/5000]      average score: 0.00
episode [800/5000]      average score: 0.00
episode [900/5000]      average score: 0.00
episode [1000/5000]     average score: 0.00
episode [1100/5000]     average score: 0.00
episode [1200/5000]     average score: 0.00
episode [1300/5000]     average score: 0.00
episode [1400/5000]     average score: 0.00
episode [1500/5000]     average score: 0.00
episode [1600/5000]     average score: 0.00
episode [1700/5000]     average score: 0.00
episode [1800/5000]     average score: 0.00
episode [1900/5000]     average score: 0.00
episode [2000/5000]     average score: 0.00
episode [2100/5000]     average score: 0.00
episode [2200/5000]     average score: 0.00
episode [2300/5000]     average score: 0.00
episode [2400/5000]     average score: 0.00
episode [2500/5000]     average score: 0.00
episode [2600/5000]     average score: 0.00
episode [2700/5000]     average score: 0.00
episode [2800/5000]     average score: 0.00
episode [2900/5000]     average score: 0.00
episode [3000/5000]     average score: 0.00
episode [3100/5000]     average score: 0.00
episode [3200/5000]     average score: 0.00
episode [3300/5000]     average score: 0.00
episode [3400/5000]     average score: 0.00
episode [3500/5000]     average score: 0.00
episode [3600/5000]     average score: 0.00
episode [3700/5000]     average score: 0.00
episode [3800/5000]     average score: 0.00
episode [3900/5000]     average score: 0.00
episode [4000/5000]     average score: 0.00
episode [4100/5000]     average score: 0.00
episode [4200/5000]     average score: 0.00
episode [4300/5000]     average score: 0.00
episode [4400/5000]     average score: 0.00
episode [4500/5000]     average score: 0.00
episode [4600/5000]     average score: 0.00
episode [4700/5000]     average score: 0.00
episode [4800/5000]     average score: 0.00
episode [4900/5000]     average score: 0.00
episode [5000/5000]     average score: 0.00

overall average score: 0.00
```

Run12: went to (far) larger NN: 512 → 256 → 64 → 16 → 2, took forever to process. Learning was up and down. Run time 6+ hours. Did pretty well, but didn't complete in a stable way.

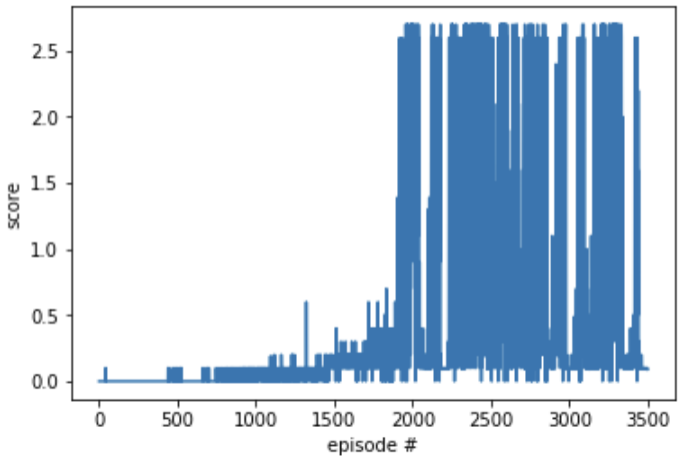
```
episode [100/5000]      average score: 0.00
episode [200/5000]      average score: 0.00
episode [300/5000]      average score: 0.00
episode [400/5000]      average score: 0.00
episode [500/5000]      average score: 0.00
episode [600/5000]      average score: 0.01
episode [700/5000]      average score: 0.00
episode [800/5000]      average score: 0.00
episode [900/5000]      average score: 0.02
episode [1000/5000]     average score: 0.01
episode [1100/5000]     average score: 0.01
episode [1200/5000]     average score: 0.02
episode [1300/5000]     average score: 0.06
episode [1400/5000]     average score: 0.10
episode [1500/5000]     average score: 0.11
episode [1600/5000]     average score: 0.19
episode [1700/5000]     average score: 0.82
episode [1800/5000]     average score: 1.29
episode [1900/5000]     average score: 1.13
episode [2000/5000]     average score: 1.74
episode [2100/5000]     average score: 0.31
episode [2200/5000]     average score: 1.41
episode [2300/5000]     average score: 0.72
episode [2400/5000]     average score: 1.06
episode [2500/5000]     average score: 0.84
episode [2600/5000]     average score: 1.24
episode [2700/5000]     average score: 1.57
episode [2800/5000]     average score: 1.19
```

Run13: Changed ‘solved’ to >2.25 (maybe 2.4 is too much to ask across an entire 100 trailing episodes). Did okay for a while, then unlearned. Timed out. Miss.

```
episode [100/3500]      average score: 0.00
episode [200/3500]      average score: 0.00
episode [300/3500]      average score: 0.00
episode [400/3500]      average score: 0.00
episode [500/3500]      average score: 0.01
episode [600/3500]      average score: 0.01
episode [700/3500]      average score: 0.00
episode [800/3500]      average score: 0.01
episode [900/3500]      average score: 0.03
episode [1000/3500]     average score: 0.06
episode [1100/3500]     average score: 0.06
episode [1200/3500]     average score: 0.06
episode [1300/3500]     average score: 0.08
episode [1400/3500]     average score: 0.09
episode [1500/3500]     average score: 0.10
episode [1600/3500]     average score: 0.12
episode [1700/3500]     average score: 0.12
episode [1800/3500]     average score: 0.17
episode [1900/3500]     average score: 0.16
episode [2000/3500]     average score: 1.11
episode [2100/3500]     average score: 0.80
episode [2200/3500]     average score: 0.95
episode [2300/3500]     average score: 1.11
episode [2400/3500]     average score: 1.56
episode [2500/3500]     average score: 1.48
episode [2600/3500]     average score: 1.41
episode [2700/3500]     average score: 1.09
episode [2800/3500]     average score: 1.15
episode [2900/3500]     average score: 0.77
episode [3000/3500]     average score: 0.95
episode [3100/3500]     average score: 0.59
episode [3200/3500]     average score: 0.59
episode [3300/3500]     average score: 1.78
episode [3400/3500]     average score: 0.86
episode [3500/3500]     average score: 0.32

overall average score: 0.50
```

Bub WOW, look how many runs were above 2.5!



What the hell?!? SO much higher average score, but never cracked 2.0, much less 2.25.

Run14: set at 2.25 again. Start time 11am PST. Learned pretty well, then bounced back down.

```
episode [100/3000]      average score: 0.00
episode [200/3000]      average score: 0.00
episode [300/3000]      average score: 0.00
episode [400/3000]      average score: 0.00
episode [500/3000]      average score: 0.02
episode [600/3000]      average score: 0.02
episode [700/3000]      average score: 0.01
episode [800/3000]      average score: 0.00
episode [900/3000]      average score: 0.01
episode [1000/3000]     average score: 0.01
episode [1100/3000]     average score: 0.00
episode [1200/3000]     average score: 0.03
episode [1300/3000]     average score: 0.05
episode [1400/3000]     average score: 0.14
episode [1500/3000]     average score: 0.12
episode [1600/3000]     average score: 0.16
episode [1700/3000]     average score: 0.75
episode [1800/3000]     average score: 0.80
episode [1900/3000]     average score: 0.23
episode [2000/3000]     average score: 0.31
episode [2100/3000]     average score: 1.67
episode [2200/3000]     average score: 0.65
episode [2300/3000]     average score: 1.20
episode [2400/3000]     average score: 0.85
episode [2500/3000]     average score: 0.08
episode [2600/3000]     average score: 0.62
episode [2700/3000]     average score: 1.78
episode [2800/3000]     average score: 1.34
episode [2900/3000]     average score: 0.69
episode [3000/3000]     average score: 0.11

overall average score: 0.39
```

Validation data across agents (solved at .5, 1.0 and 2.0):

```
actor file:  km_actor_a.pth
critic file: km_critic_a.pth
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.045     player volleys: 1.1
max agent score: 0.200   averaged agent score: 0.195     player volleys: 6.1
max agent score: 0.100   averaged agent score: 0.045     player volleys: 2.1
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.095     player volleys: 2.6
max agent score: 0.200   averaged agent score: 0.145     player volleys: 5.1
max agent score: 0.100   averaged agent score: 0.095     player volleys: 4.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.100   averaged agent score: 0.095     player volleys: 2.8
max agent score: 0.200   averaged agent score: 0.145     player volleys: 5.0
max agent score: 0.100   averaged agent score: 0.095     player volleys: 3.9
max agent score: 0.090   averaged agent score: 0.045     player volleys: 2.1
max agent score: 0.100   averaged agent score: 0.095     player volleys: 3.8
max agent score: 0.100   averaged agent score: 0.045     player volleys: 2.1
max agent score: 0.100   averaged agent score: 0.045     player volleys: 2.1
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
average across test run: 0.969   average volleys: 27.2

actor file:  km_actor_b.pth
critic file: km_critic_b.pth
max agent score: 2.490   averaged agent score: 2.445     player volleys: 65.9
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 1.590   averaged agent score: 1.545     player volleys: 41.6
max agent score: 0.600   averaged agent score: 0.495     player volleys: 14.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 0.200   averaged agent score: 0.145     player volleys: 3.8
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 1.600   averaged agent score: 1.595     player volleys: 43.3
max agent score: 0.600   averaged agent score: 0.545     player volleys: 17.4
max agent score: 0.590   averaged agent score: 0.545     player volleys: 15.4
max agent score: 2.600   averaged agent score: 2.550     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
average across test run: 2.061   average volleys: 56.6

actor file:  km_actor_c.pth
critic file: km_critic_c.pth
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
max agent score: 2.600   averaged agent score: 2.600     player volleys: 71.5
max agent score: 2.700   averaged agent score: 2.650     player volleys: 71.5
average across test run: 2.635   average volleys: 71.5
```

Run15: set at 2.25 again (I’m not giving up yet).

```
episode [100/3000]      average score: 0.01
episode [200/3000]      average score: 0.00
episode [300/3000]      average score: 0.00
episode [400/3000]      average score: 0.00
episode [500/3000]      average score: 0.00
episode [600/3000]      average score: 0.02
episode [700/3000]      average score: 0.00
episode [800/3000]      average score: 0.00
episode [900/3000]      average score: 0.00
episode [1000/3000]     average score: 0.00
episode [1100/3000]     average score: 0.00
episode [1200/3000]     average score: 0.00
episode [1300/3000]     average score: 0.02
episode [1400/3000]     average score: 0.05
episode [1500/3000]     average score: 0.09
episode [1600/3000]     average score: 0.13
episode [1700/3000]     average score: 0.37
episode [1800/3000]     average score: 0.31
episode [1900/3000]     average score: 1.04
episode [2000/3000]     average score: 0.15
episode [2100/3000]     average score: 1.07
episode [2200/3000]     average score: 0.98
episode [2300/3000]     average score: 0.89
episode [2400/3000]     average score: 0.89
episode [2500/3000]     average score: 1.21
episode [2600/3000]     average score: 1.11
episode [2700/3000]     average score: 0.86
episode [2800/3000]     average score: 1.28
episode [2900/3000]     average score: 1.40
episode [3000/3000]     average score: 1.07

overall average score: 0.43
```

Run16: try again? Set ‘solved’ at 2.25.

Didn’t converge to 2.0+. Got to 1.68 max.