

Cat and Mouse Improvement

Zach Butler, Shalini Kusuru, Markus Schultz, Reddi Babu Gundugallu

November 29, 2018

Contents

In the Boxes World in the cat and mouse reinforcement learning applet we attempted several modifications of the existing in order to gather performance gains. Most of the attempts were failures. We will briefly highlight them below.

First we ran the simulation with no modifications and all user-accessible variables left unchanged.

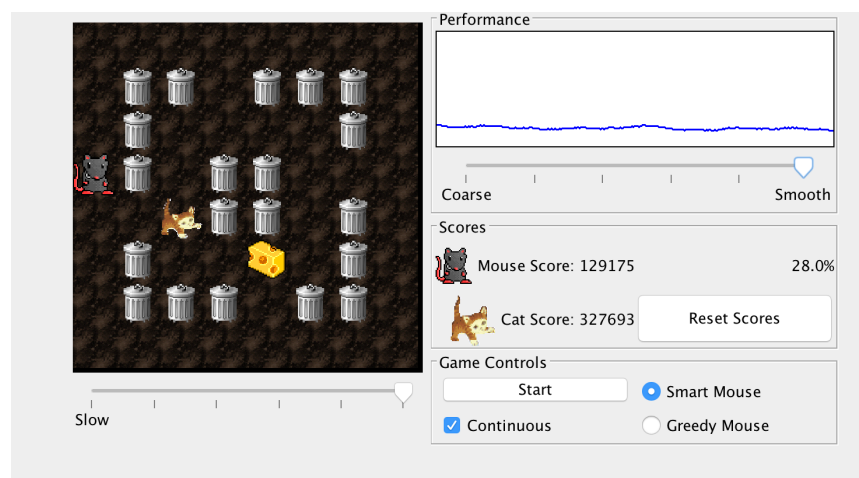


Figure 1: Performance Before Alteration

```

public double calcReward() {
    double newReward = 0;
    if ((mx==chx)&&(my==chy)) {
        mousescore++;
        newReward += cheeseReward;
    }

    if ((cx==mx) && (cy==my)) {
        catscore++;
        newReward -= deathPenalty;
    }
    //if ((mx==hx)&&(my==hy)&&(gotCheese)) newReward += 100;

    return newReward;
}

```

Figure 2: Original Code

Our first approach was to give a reward to the mouse for maintaining specific distances from the cat. However that solution offered no benefit. Several values for reward given and distance maintained were used, but no change was measured.

Next we attempted to remove invalid actions from the q-table. There were improvements, but as this approach was not our original idea and performed no better than our original idea, we believe that offering details in this report is of little use.

Finally we used the solution from previous worlds that yielded good results. This solution again performed very well.

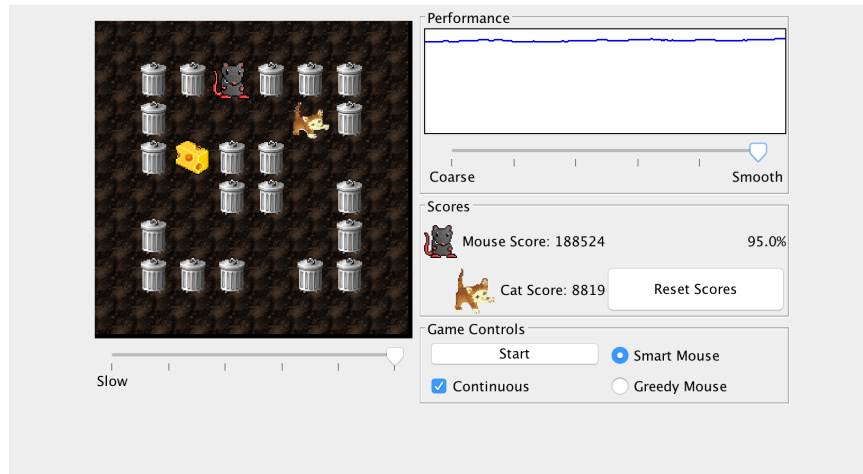


Figure 3: Performance After Alteration

The code below was inserted into `calcReward`.

```
if ((mx != cx) && (my != cy)) {  
    newReward += 10;  
}
```

Figure 4: Code Revision

Again, giving an arbitrary reward less than the cheese reward for every safe move yields the best results.