Modeling Mice Performance on the Morris Water Maze Test

Benjamin Goebel

Tufts University, Medford, MA

Abstract

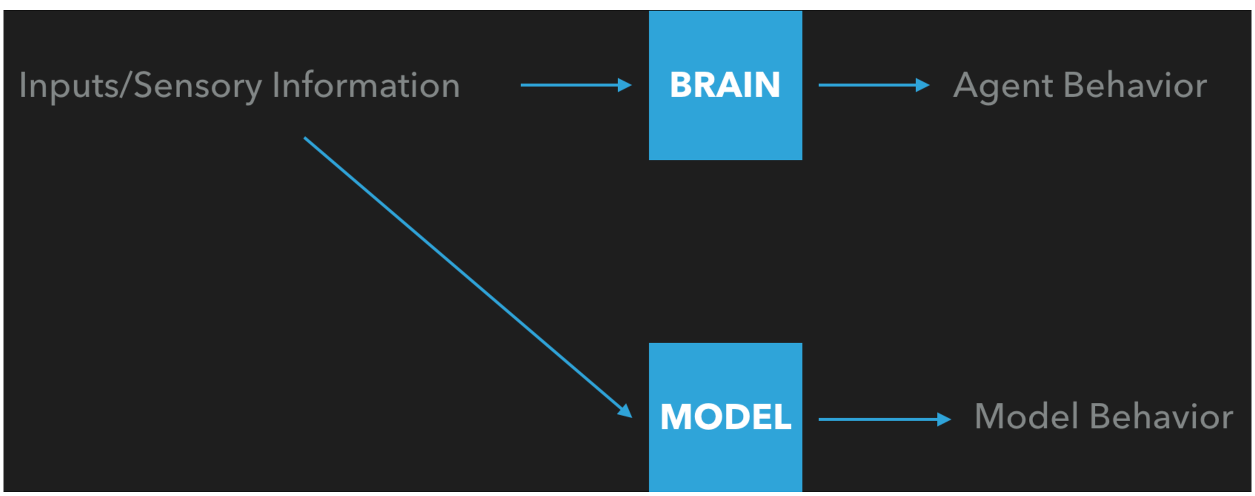
This work seeks to model mouse performance on a spatial memory task, the Morris water maze test. Spatial memory is the ability to remember locations in space. The Morris water maze test and mouse performance on the Morris water maze test is described in Morris (1981). In the Morris water maze test, mice seek to find a hidden platform placed slightly below the surface of water, in a water tank, where the water is opaque. Q-learning with a softmax, a reinforcement learning process, is used to model the behavior. The model produces similar results, in a 7 x 7 square environment, to the performance of mice in Morris (1981), suggesting reinforcement learning as a possible cognitive mechanism for spatial memory learning, and possibly all learning, in mice. Because the model is limited to a 7 x 7 square environment, we discuss how this model could be adapted to work in different environments, citing research on deep learning. Adapting this model leaves us with a more general model and explanation for spatial memory learning in mice.

Keywords: spatial memory, modeling, exploration, exploitation, Q-learning, softmax

1. Modeling Behavior in Psychology

Modeling behavior in psychology seeks to provide a mechanism for an animal's behavior under assumptions. Modeling behavior in psychology simplifies the task. A task can be broken down into inputs, a mechanism, and outputs. The inputs are sensory information that is available to the agent, while the agent completes the task (i.e. your five senses). A mechanism is a cognitive mechanism for how the agent's brain could complete the task. Outputs are the agent's resulting behavior. Consequently, if we take the inputs that are available to the agent, while the agent completes the task and swap the agent's brain for a suggested mechanism to complete the task, we can compare the outputs of our mechanism with how the agent actually performs during this task; this comparison would determine how well the model predicts the agent's behavior. The following ideas are illustrated, below, in Figure 1.

Figure 1



The goal of the model is to have the model behavior be roughly equivalent to the agent behavior. And if this condition is met, then we have successfully created a cognitive mechanism to explain the agent's behavior. This does not mean that we have determined how the agent's brain completes the task; however, in building this model, we can learn more about how the agent's brain could complete this task.

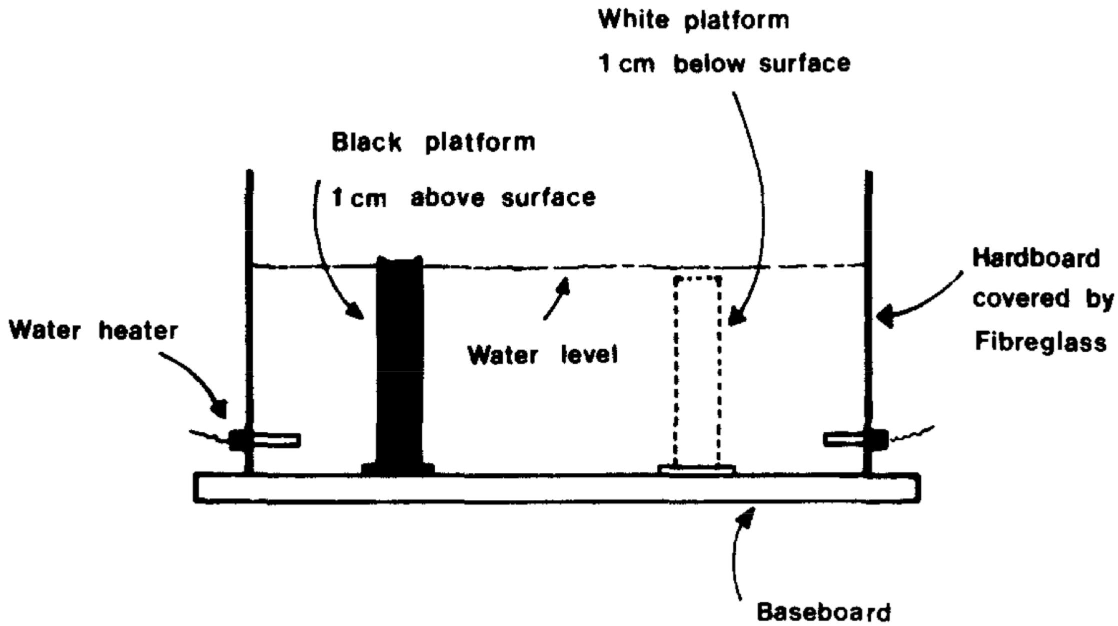
2. The Morris Water Maze Test

The goal of our model is to model the behavior of mice in the Morris water maze test. The Morris water maze test is delineated by Morris (1981). The Morris water maze test is used to assess the spatial memory of mice. In this test, a mouse is placed in a water tank with a platform located in the water tank. The mouse's goal is to find the platform (Morris, 1981, p. 240).

In this test, there are four experimental groups: Cue + Place, Place, Cue-only, and Place-Random. The mice are trained in these experimental groups. Group Cue + Place is trained on a black platform that is placed slightly above the surface of the water; the platform's location remains fixed throughout training. Group Place is trained on a white platform that is placed slightly below the surface of the water; the platform's location remains fixed throughout training. This experimental group is the main group of interest of these four experimental groups. Additionally, Group Cue-only is trained on a black platform that is placed slightly above the surface of the water; the platform's location changes from trial-to-trial, throughout training. Group Place-Random is trained on a white platform that is placed slightly below the surface of the water; the platform's location changes from trial-to-trial, throughout training (Morris, 1981, p. 242).

For the experimental groups where a white platform is placed below the surface of the water, the platform is tall enough, such that, when a mouse rests on the platform, the mouse's head is above the water surface. This height allows mice to breathe easily when resting on the white platform. Further, a white, milky substance is added to the water to render the water opaque; such that, the mouse cannot see the white platform, since the white platform is below the surface of the water; however, the mouse can see the black platform, since the black platform is above the surface of the water (Morris, 1981, pp. 240-241). Consequently, to find the platform, the mouse has to search for it. Adding the milky substance removes any local cues for the mouse to rely on. Local cues can be thought of as items in the water maze that can be used to help guide the mouse to the platform. Instead, all that is available to the mouse is distal cues, or cues located in the room, in which, the water maze is situated. When Morris (1981) runs the experiment, a door, a wall, a cupboard, a window, and shelves were all located in the room for the mice to see (Morris, 1981, pp. 239-241). Figure 2, shown below, is a schematic that illustrates the water maze with both platforms: the black platform, slightly above the surface of the water, and the white platform, slightly below the surface of the water.

Figure 2



(Morris, 1981, p. 241)

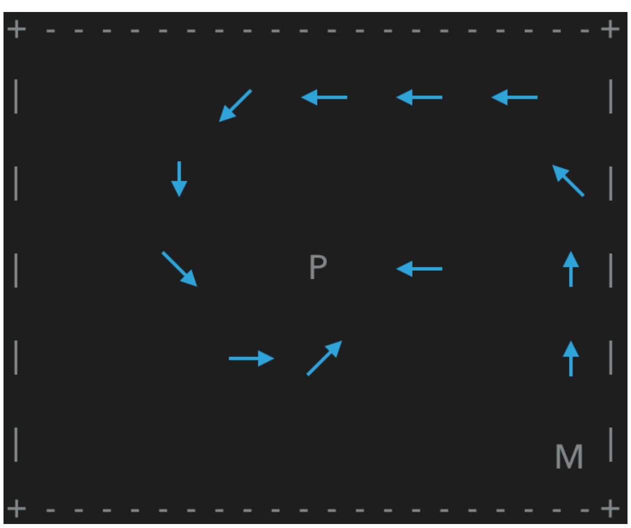
Groups Cue + Place and Cue-only are used to draw attention to group Place; Morris (1981) compare the mice's performance of group Place to the mice's performance of the two previously mentioned groups to demonstrate that mice, with relying on distal and no local cues, can develop a spatial bias to an area of the water maze and move there almost as quickly as mice in experimental groups where there exist both distal and local cues (Morris, 1981, p. 252). Morris (1981) compares the performance of mice in group Place-Random to the performance of mice in the three other experimental groups to indicate that the performance of mice, in group Place-Random, is severely impaired, in comparison to the performance of mice in the three other experimental groups. This is the case because for group Place-Random, there does not seem to be any local cues (the mouse cannot see the platform) and the location of the platform changes from trial-to-trial (Morris, 1981, p. 246).

3. Understanding The Task

The model, delineated in this paper, seeks to model the task a mouse faces in group Place of Morris (1981). Before presenting the model, it is important to consider and understand the task that is being modeled. Through understanding the task, we can see the challenges a mouse faces in developing a spatial bias with only distal and no local cues. Further, after the model is presented in the next section of this paper, one can consider how the model approaches solving these challenges. The images in Figures 3, 4, and 5 will help us understand the task.

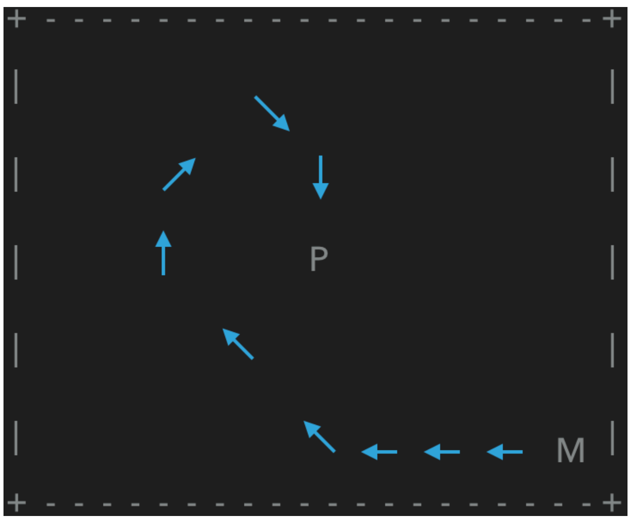
In these images a mouse, M, is taking paths, in an attempt to find the hidden platform, P. If the mouse, in Figure 3, has no prior knowledge and is attempting to find the platform, perhaps it will take a roundabout path to find the platform.

Figure 3



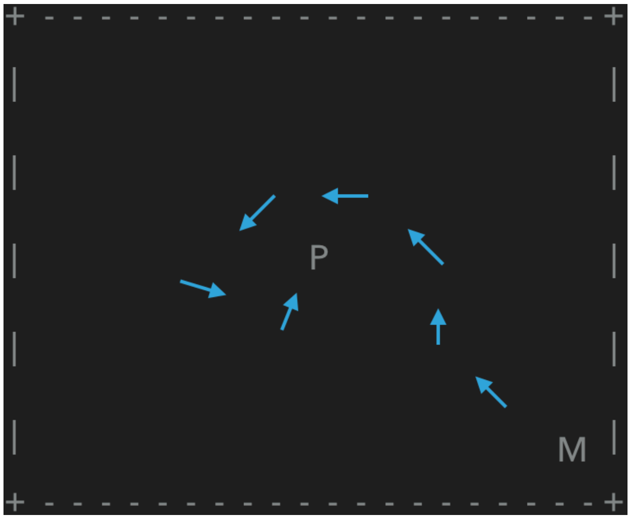
After the mouse has found this path to the platform, in Figure 3, let's assume this path is all the knowledge the mouse has on how to find the platform in the water maze. Consequently, given this task again, in Figure 4, the mouse can rely on the path it takes in Figure 3, or if the mouse believes it can find a quicker path, it can explore the water maze more, taking a new path to find the platform. Let's assume the mouse chooses to explore the water maze more, as shown in Figure 4.

Figure 4



After the mouse has found this path to the platform, in Figure 4, let's assume this path and the path the mouse takes in Figure 3 are all the knowledge the mouse has on how to find the platform in the water maze. Consequently, given the task one more time, in Figure 5, the mouse can rely on the paths it takes in Figures 3 and 4, or if the mouse believes it can find a quicker path, it can explore the water maze even more, taking a new path to find the platform. Let's assume the mouse, again, chooses to explore the water maze, as shown in Figure 5.

Figure 5



These paths, in Figures 3, 4, and 5, explore and gain knowledge about the water maze. And the problem that arises, in this example, is the exploration-exploitation dilemma. The exploration-exploitation dilemma is the conflict the agent faces, when beginning subsequent trials. In this dilemma, the agent can either exploit its knowledge of the system, by taking the quickest path, of which, it knows to find the platform, or the agent can explore the system, moving in new directions and visiting previously unvisited locations, in the water maze, with the possibility of finding a quicker path to the platform than the agent previously knew existed.

4. The Model

The model uses Q-learning, a reinforcement learning process, to encode information about the water maze, and it uses a softmax to make decisions about where to move in the water maze.

MODELING MICE PERFORMANCE ON MWM

4.1 Q-learning

Q-learning is a reinforcement learning process used to encode information about systems. It assigns q-values to states to indicate the utility of each state. The equation used to assign q-values to states is shown, below, in Figure 6.

Figure 6

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma \max_{a'} Q(s',a') - Q(s,a)]$$

This equation is used to estimate the q-value, utility, of a state that is visited in the system by an agent. In this equation, Q(s, a) is the q-value for the current state; α is the learning rate, how quickly the agent learns the utility of the states in the system; r is the reward for transitioning from state-to-state. In our model, this variable will be set to zero for all states. In the equation, γ is the discount factor; its value is set to be between zero and one. It is used to discount the q-values of states. Q-values of states farther away from the goal state are discounted more than q-values of states closer to the goal state. This is the case because states closer to the goal state have higher utility values than states farther away from the goal state. Additionally, in the equation, the term, $\max_{a'} Q(s', a')$, is the largest q-value in an adjacent state to the current state. This term is discounted, by being multiplied by γ (remember $0 < \gamma < 1$) because adjacent states are one state away from current states.

With this brief introduction to Q-learning, let's turn back to the Morris water maze test. Now, imagine we represent the water maze and its respective states as a grid, where each cell, in the grid, corresponds to one state and its respective q-value. In this grid, we could set the goal state to have an initial q-value of 100 (an arbitrary large number) and all other states to have an initial q-value of 0. Further, if we repeatedly visit all states and apply the q-learning equation, shown in Figure 6, to each of these states, the resulting grid of q-values will be a gradient; larger q-values will be closer to the location of the platform, and smaller q-values will be father from the location of the platform. This gradient is terrific because we can, now, rely on these q-values to find the hidden platform. However, to form this gradient we need to learn the q-value of all possible states because the q-value of each state informs the q-value of its adjacent states. Consequently, as in the beginning trials of the Morris water maze test, when the model is updating the q-value of the current state and does not know the q-values of all adjacent states, the model may inaccurately estimate the q-value of the current state to be lower than it actually is; it is only through further exploration of the adjacent states that this inaccurate estimate can be corrected.

4.2 Softmax

As has been shown in previous sections, we can find paths to the platform and form estimates about the utility (q-value) of each state; however, this path tends to not be the shortest path and these utilities tend to be inaccurately, low estimates. It has been concluded that some exploration (i.e. moving to previously unvisited states) is necessary to help find shorter paths and improve the accuracy of these utility (q-value) estimates. What is needed is a mechanism to balance exploration, visiting new states, with exploitation, taking advantage of accumulated knowledge, q-value estimates. The softmax seeks to handle this problem. The softmax examines all possible choices in a decision and assigns probabilities to each choice. The equation the softmax uses to assign a probability to each choice is shown, below, in Figure 7.

Figure 7

$$P(a_i) = \frac{e^{Q(a_i)/T}}{\sum_k e^{Q(a_k)/T}}$$

In this equation, $P(a_i)$ represents the probability of moving to an adjacent state, a possible choice; $Q(a_i)$ represents the adjacent state's q-value, and T represents the computational temperature. The computational temperature modulates the level of exploration and exploitation the model exhibits; as the computational temperature approaches infinity the model favors exploration, and as the computational temperature approaches zero, the model favors exploitation.

Returning back to the Morris water maze test, imagine we ran the model on the Morris water maze test for five trials. In the first trial, we could set the computational temperature to be a very large number to encourage exploration, and after each trial, we could decrement the computational temperature; such that, as the model completes the fourth and fifth trials, the computational temperature is very low, encouraging exploitation.

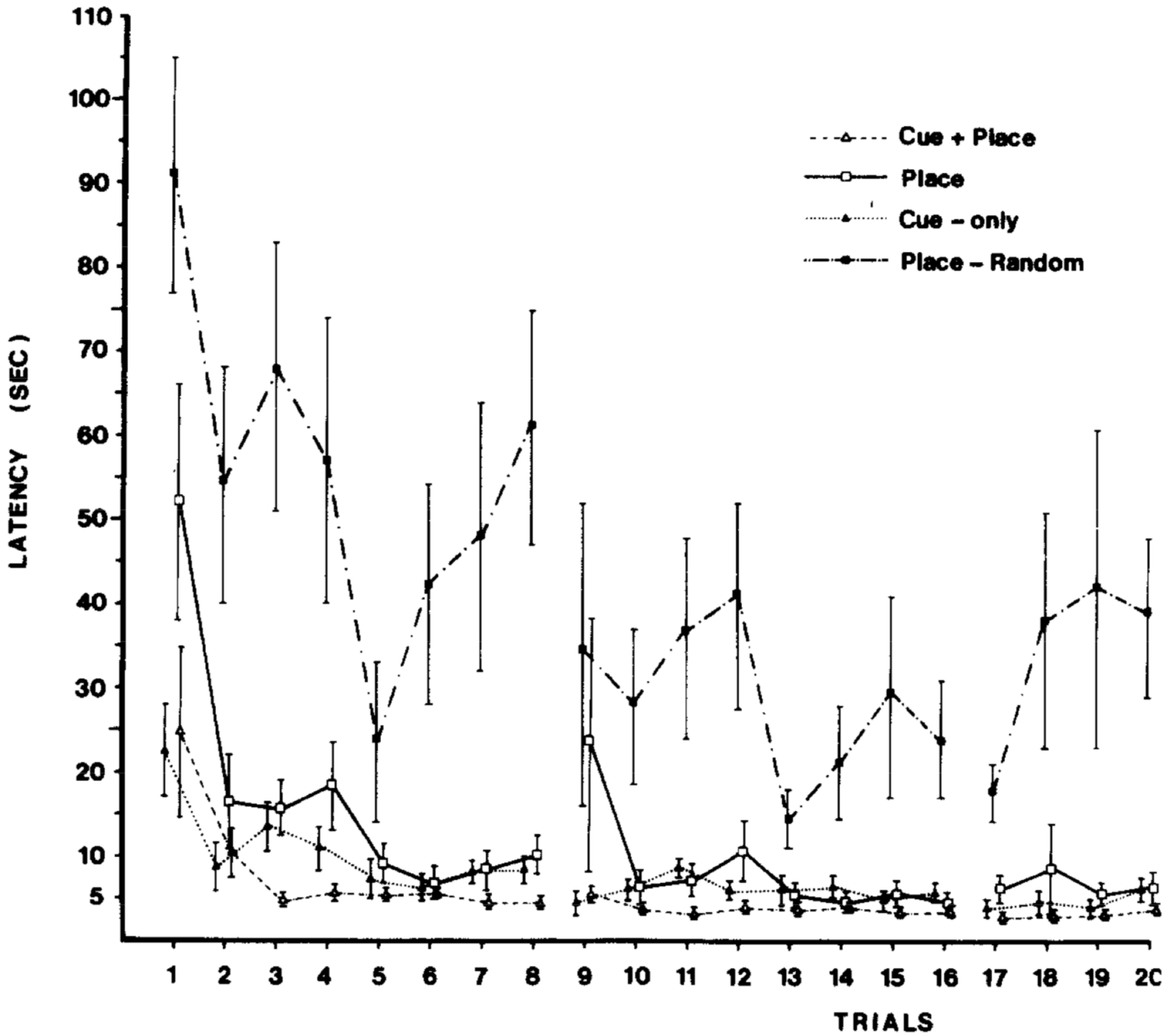
Reflecting on this strategy, in encouraging exploration in the first few trials, the model will be able to take different paths to the platform and accumulate knowledge (q-value estimates) about many states. In accumulating this knowledge, the goal is to establish the gradient of q-values, as discussed in section 4.1, as quickly as possible. Once, this gradient is established, the model can exploit its knowledge (q-value estimates) of the water maze. Lastly, in decrementing the computational temperature, we will need another parameter, MOD_T, to decrement the computational temperature after each trial. This parameter will be set to a value between zero and one, so that after each trial, T can be set to the product of itself and MOD_T.

5. Results

5.1 Training Results

Returning to the four experimental groups from Morris (1981), these four groups train for three days. On the first two days, eight training trials are performed on each group, each day, and on the following day, four training trials are performed on each group. For all groups the start locations change from day-to-day, and for groups Cue-Only and Place-Random the end location of the platform changes. For each trial, regardless of the experimental group, the mouse starts in the center of one of the sides of the water maze, and for each trial, regardless of the experimental group, the platform is placed in one of the corners of the water maze. For all groups, across all training trials, latency is measured as the amount of time it takes the mouse to find the platform (Morris, 1981, pp. 242-243). Figure 8, below, shows the average latency for each group for each trial. Again, remember that the main group of interest is group Place.

Figure 8

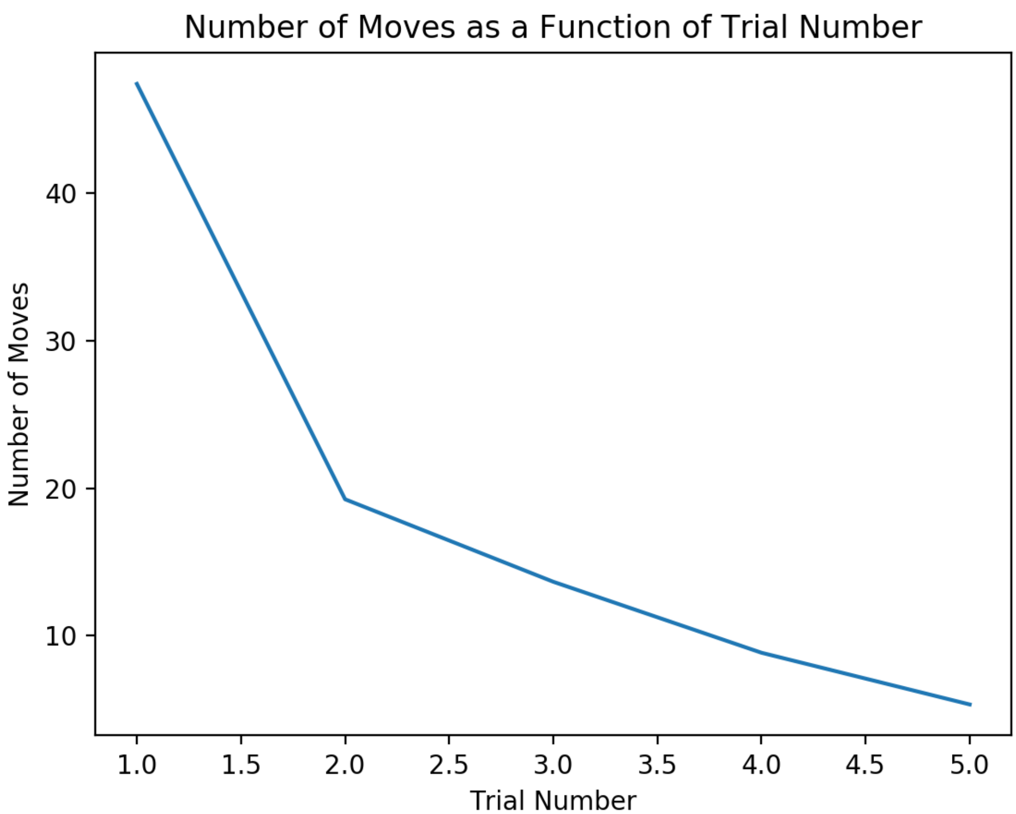


(Morris, 1981, p. 246)

On the first training day, over the first 8 trials, group Place severely improves its latency results from ~50 seconds, on trial 1, to ~10 seconds, on trial 8. On the second training day, group Place improves its latency from ~25 seconds, on trial 9, to 5-10 seconds, on trial 16. And on the last training day, group Place keeps its latency results fairly constant in the 5-10 seconds range. A possible explanation for the uptick in latency on trial 9 is that the amount of time between the last trial on the previous day and the first trial on the following day is greater than the amount of time between trials on the same day. However, the amount of time between trials is not mentioned in Morris (1981).

In response to this data, the model was run for five trials over 1,000 iterations, with the computational temperature set, initially, to 100.0, TEMP_MOD set to 0.55, the discount factor set to 0.9, and the learning rate set to 0.9. The number of moves the model takes, on average, to find the platform, for each trial, was measured. The results are shown, below, in Figure 9. In the model, the water maze is represented as a 7 x 7 board, with the platform placed in the center of the board, and the mouse starting in a corner of the board. Over the trials, the mouse's starting position alternates between all four corners of the board, and the platform's location remains fixed.

Figure 9



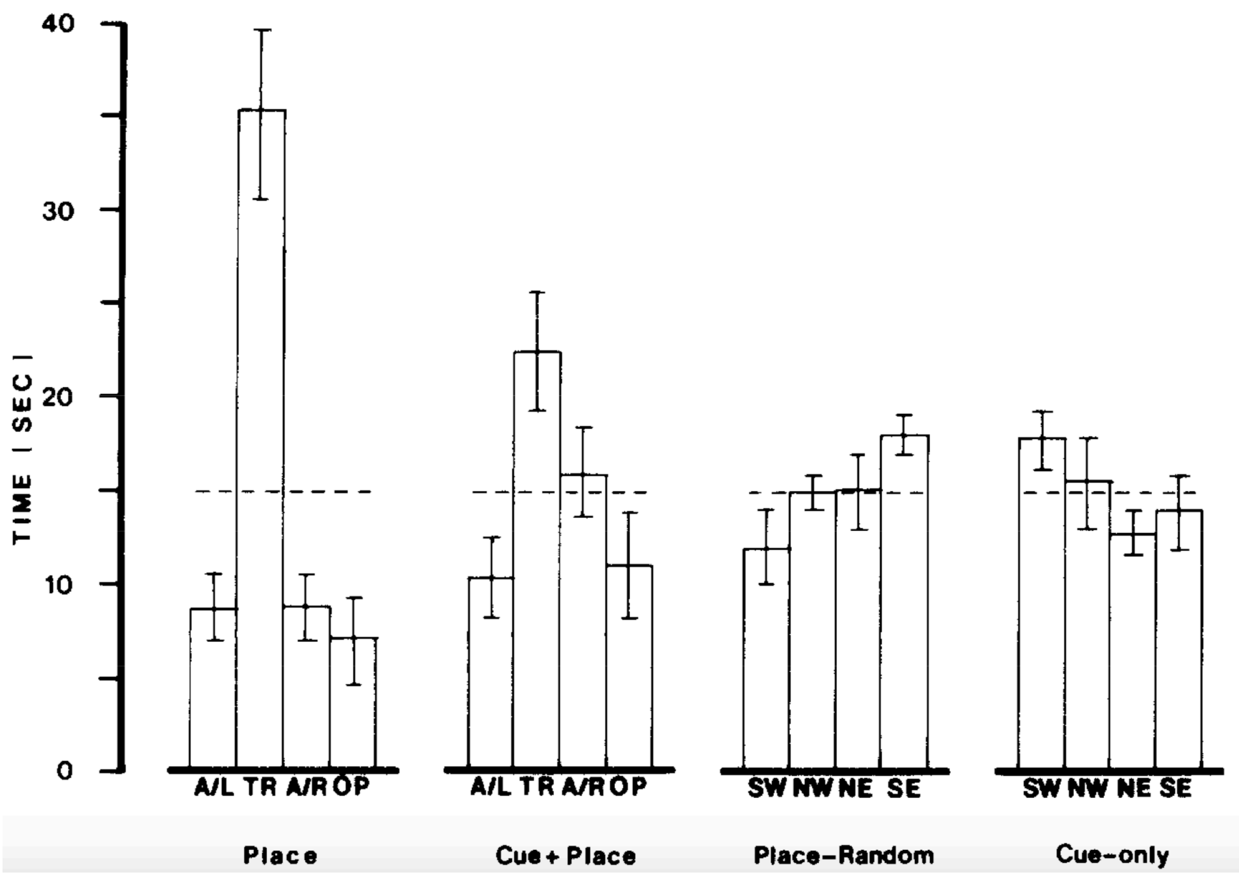
Similar to the mice in group Place, the model starts at a large latency of ~50 moves, in trial 1, and severely decreases its latency to under 10 moves by trial 5.

5.2 Spatial Bias Results

Following the 20 training trials, Morris (1981) divides each of the four experimental groups into two groups, assigning one group to Test A and one group to Test B. Consequently, all four experimental groups participate in each test, with each group's size being half of its original size. Test A is discussed in this section, and Test B is discussed in the following section.

Test A is a probing trial. In Test A, the mice are placed in the water maze with the platform removed from the water. The mice are left in the water maze for 1 minute, and the amount of time, in seconds, the mouse spends in each quadrant of the water maze is measured (Morris, 1981, p. 243). The results from Test A are shown in Figure 10, below.

Figure 10

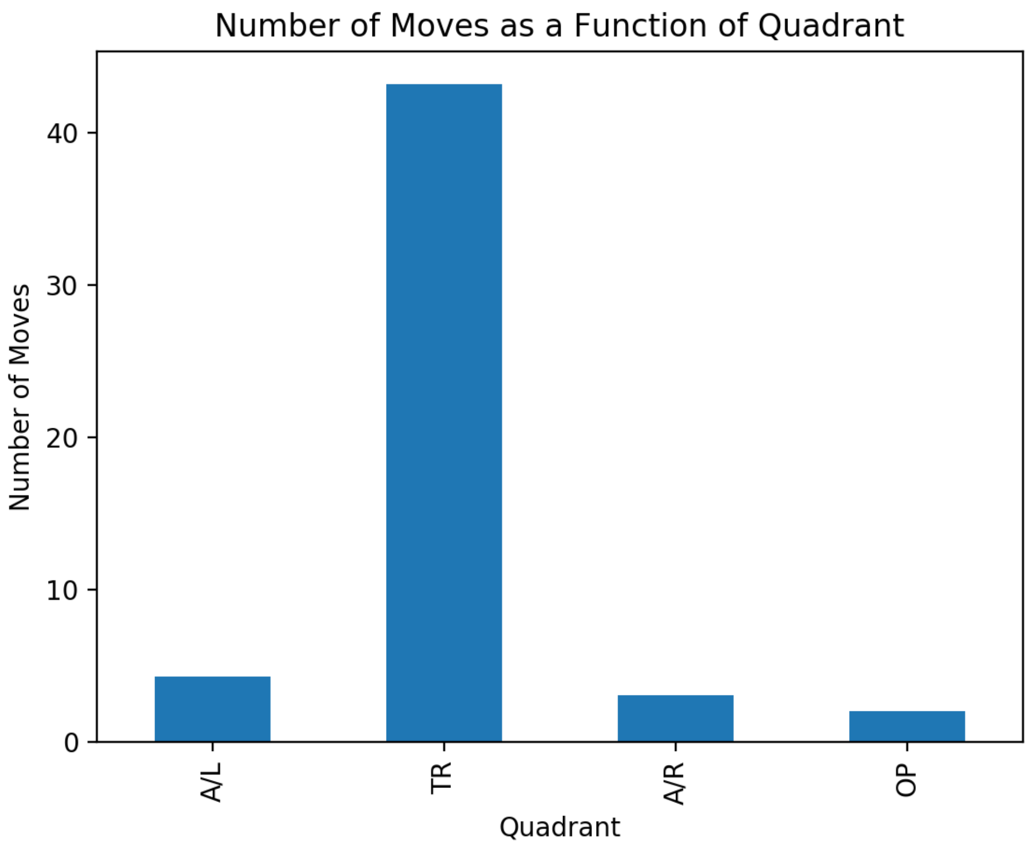


(Morris, 1981, p. 250)

In this graph, "TR" stands for target quadrant, where the platform is placed before being removed. "A/L" stands for adjacent to the left of the target quadrant. "A/R" stands for adjacent to the right of the target quadrant, and "OP" stands for opposite of the target quadrant. The portion of the graph of interest is the leftmost grouping of four columns, which is the data corresponding to group Place's results. As shown by the graph, group Place demonstrates a significant spatial bias to the target quadrant, with the mice in this group spending much more time in the target quadrant than they spend in any other quadrant of the water maze.

In response to this data, the model was run for five trials, with the computational temperature set, initially, to 100.0, TEMP_MOD set to 0.55, the discount factor set to 0.9, and the learning rate set to 0.9. After the five trials, the platform was removed from the water maze, and the model was allowed to make 60 moves in the water maze. The number of moves the model takes in each quadrant was measured. This process was repeated for 1,000 iterations. The results are shown, below, in Figure 11. The results show the average number of moves the model makes in each quadrant. Again, for the model, the water maze is represented as a 7 x 7 board. This time, the platform is placed in the corner of the board, and the mouse starts in one of the other three corners of the board. The mouse alternates between these three corners, during training and probing.

Figure 11



Similar to the mice in group Place, the model demonstrates a spatial bias to the target quadrant, after training, with the model spending much more time in the target quadrant than it spends in any other quadrant. The spatial bias, in the model's data, appears to be more severe than the spatial bias that is demonstrated by group Place in Morris (1981).

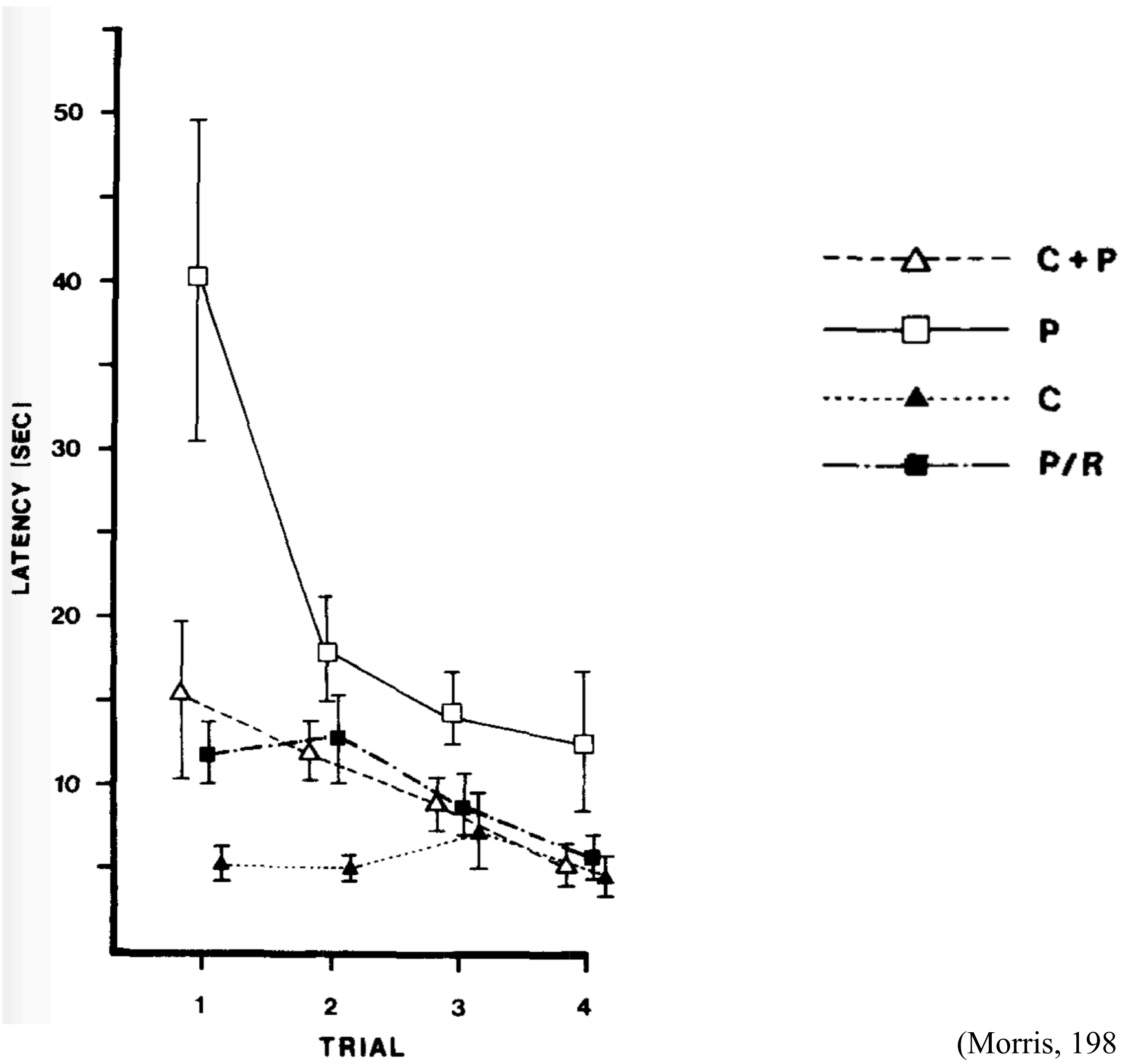
5.3 Forgetting a Spatial Bias

As mentioned above, following training, Morris (1981) divides each of the four experimental groups into two groups, assigning one group to Test A and one group to Test B. Test

A and its results are discussed in the previous section. Test B and its results are discussed in this section.

In Test B, Morris (1981), for group Place, places the platform in a new location in the water maze (Morris, 1981, p. 243). The mice, in group Place, are trained on this new platform location, after being trained on the previous platform location. The results are shown, below, in Figure 12.

Figure 12



(Morris, 1981, p. 251)

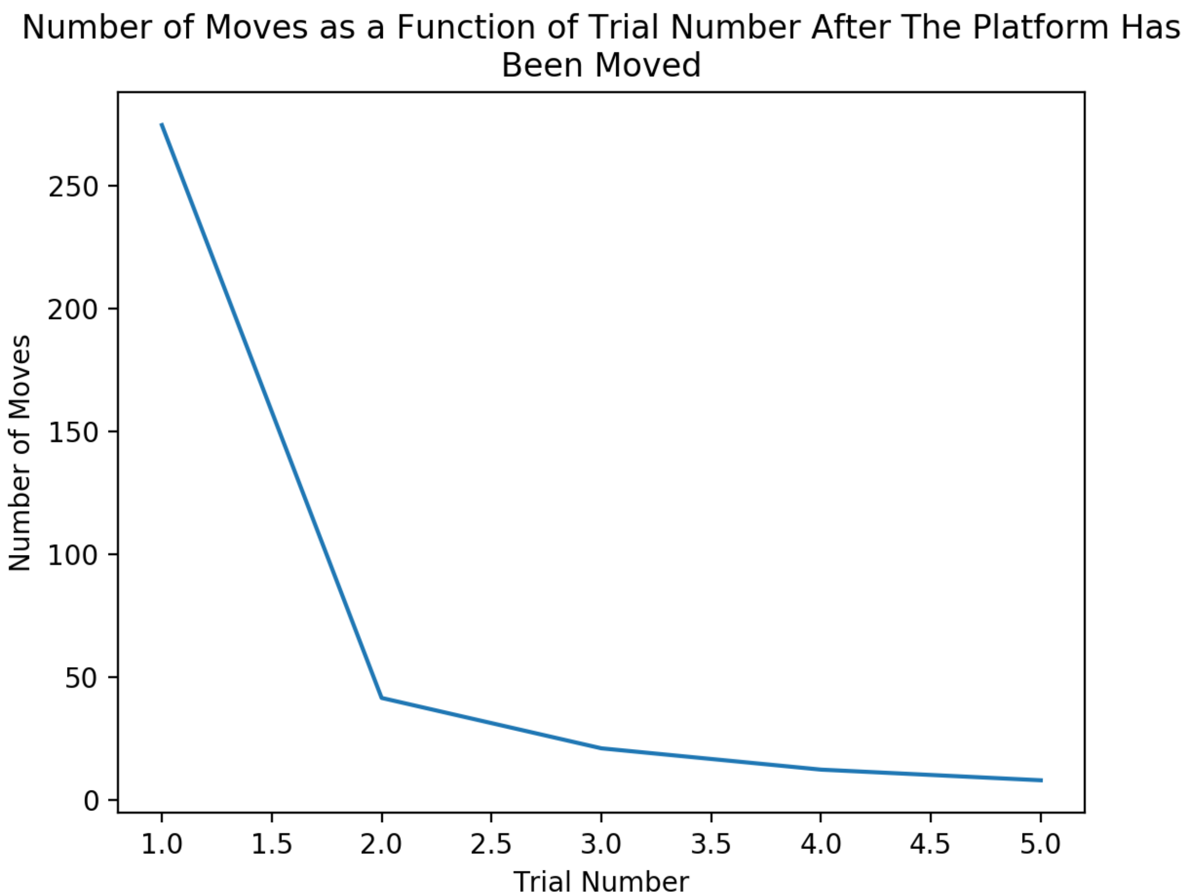
In the graph, the data of interest is the line of square points; this is the data for group Place. As shown by the graph, group Place, after the platform is moved, initially, struggles, taking ~40 seconds to find the platform in trial 1. However, group Place learns the new platform location, and by trial 4, group Place can find the new platform location in almost 10 seconds.

In response to this data, the model was given the ability to forget a spatial bias. For the model to forget a spatial bias, during a trial, it has to have moved a certain number of times without finding the platform, and it has to have been trained on a constant platform location for a specified number of trials.

The model forgets its spatial bias by resetting the computational temperature back to its initial value. Resetting the computational temperature allows the model to explore, again, in search of the new platform location. When the model finds the new platform location, it updates the q-values of the visited states, after the trial, as it normally does. However, in assigning the q-value to the state of the new platform location, this state is given a q-value twice as large as the q-value of the state of the previous platform location. Making the q-value of the state of the new platform location twice as large as the q-value of the state of the old platform location is arbitrary; the most important fact is that the q-value of the state of the new platform location is significantly larger than the q-value of the state of the old platform location. This fact allows a new gradient to form, where states closer to the new platform location are assigned relatively large q-values, and states farther away from the new platform location are assigned relatively small q-values.

With its ability to forget a spatial bias, the model was trained on a platform location for five trials, and, then, the model was trained on a new platform location for five trials. The model was run for 1,000 iterations. The computational temperature was set, initially, to 100.0, TEMP_MOD was set to 0.55, the discount factor was set to 0.9, and the learning rate was set to 0.9. For the platform to be moved, the model needed to have made at least 37 moves without finding the platform, and the model needed to have been trained on a constant platform location for at least three trials. The results are shown, below, in Figure 13. The results show the average number of moves the model takes for each trial, after the platform has been moved. Again, the model represents the water maze as a 7 x 7 board. The mouse begins each trial in a corner, and the platform is placed in another corner of the board. Over the trials, the mouse's starting position alternates between all corners where the platform is not located.

Figure 13



From the results, the model significantly struggles to find the platform on the first trial. After this trial, the model is able to learn the platform location on subsequent trials. The results of this model are similar to the results shown in Morris (1981) in Figure 12. The one difference is the severe number of moves it takes for the model to find the platform, initially.

The model's ability to forget a spatial bias is accounted in the data shown in the previous two sections. In section 5.1, there does not seem to be a reason for the model to forget its spatial bias; however, the model has the capability to, if it, unlikely, fails to learn the platform location, in the first few trials. In section 5.2, the model being able to forget its spatial bias is a significant detail to the test. Because the platform is removed from the water maze, if the model were not able to forget its spatial bias, it would be stuck in the target quadrant for the whole test. This result would not fit the data shown in Morris (1981) well. Since the model is able to forget its spatial bias, it can visit the other quadrants more frequently to produce results that are more similar to the results shown in Morris (1981).

6. Discussion

The model's results, illustrated in this paper, demonstrate that the model, for a square water maze of a specified size, can establish and forget a spatial bias, like mice do in Morris (1981). As has been discussed, the model achieves these results through representing the environment as a grid of q-values, utility values, and it uses a softmax that, initially, encourages exploration, and, finally, encourages exploitation to make decisions to optimize performance in the water maze. We say the model performs well in square water mazes because the model was run on a square water maze to collect the data shown in this paper; this is an important point. This model is representing a specific environment of a specific size. The model's representation of the environment dictates which environments, in that, the model can perform well. Here, the model's representation is a grid of 7 x 7 q-values. Consequently, the model, delineated in this paper, models water mazes that are 7 x 7 squares well; it is a shortcoming of the model that it can perform well in square environments and not in circular environments, like the water maze in Morris (1981). To model circular environments, the model's representation would need to be changed to a circular grid, instead of a square grid. Moreover, the underlying issue here is that the model only has one representation, so it can only model one type of environment well. It is reasonable to believe that mice can perform well in spatial memory tasks in environments of all shapes and sizes. As a result, for this model to perform as well as mice perform on spatial memory tasks, the model needs a more general representation of the environment.

Mnih et al. (2015) create this general representation. Mnih et al. (2015) built a learning model to play 49 different Atari games at high performance. Mnih et al. (2015) did this by feeding sensory input, pixel data, from Atari games, into neural networks to generate q-values. These q-values represent states in the games. Consequently, it seems plausible that this same neural network architecture could be used to generate q-values for the model described in this paper. Then, the model could use a softmax that, again, favors exploration, initially, and favors exploitation, finally, to optimize performance. The model would, then, be able to perform well on the Morris water maze test in all sorts of environments, where the water maze is any type of shape or size.

With this modification to the model, we, now, have a model that could be a cognitive mechanism for how mice perform spatial memory tasks. The one downside to including a neural network to any suggested cognitive mechanism is that neural networks require a large amount of training to perform a task well. As a result, the model would need much more training than a mice needs to perform this spatial memory task well. In the future, researchers should continue to investigate general reinforcement learning representations with the hope of finding a general representation that requires less training to perform well.

With a large amount of training, the modified model will perform well. And if a general reinforcement learning representation is found that requires less training, the model will perform even better.

There is a significance to the fact that a reinforcement learning model can model the performance of mice on a spatial memory task. A reinforcement learning model can learn all sorts of behavior. So, it is very possible that the cognitive mechanism a mouse uses to learn to perform any activity well might be the same cognitive mechanism that a mouse uses for spatial memory learning; this could mean that mice have one cognitive mechanism for learning.

In studying and considering models for mice behavior on spatial memory tasks, one might wonder how similar the cognitive mechanisms of mice are to that of human beings. And, the hope is that the cognitive mechanisms are very similar, so that if one can explain the cognitive mechanisms in mice's brains, one can explain some of the cognitive mechanisms in human brains. However, regardless of how similar the cognitive mechanisms are, we still get to learn about how an animal's brain functions, which, overall, informs our understanding of how brains function. Thus, in studying and modeling how mice complete spatial memory tasks, we are learning how some brains complete spatial memory tasks.

Bibliography

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A.A., Veness, J., Bellemare, M.G., … Hassabis, D. (2015). Human-level control through deep reinforcement learning. *Nature*, *518*(7540), 529-541. doi:10.1038/nature14236

Morris, R.G. (1981). Spatial localization does not require the presence of local cues. *Learning and Motivation*, *12*(2), 239-260. https://doi.org/10.1016/0023-9690(81)90020-5

Appendix

The model can be accessed at https://github.com/bengoebel/WaterMazeTest