Navigation Policy Development using Imitation Learning From Rodent Foraging Data

Annmarie Hashimoto[1,2], Xuelei Chen[2], Vittorio Giammarino[2], Ioannis Paschalidis[2]
[1]American School In Japan, 1-1-1 Nomizu, Chofu-shi, Tokyo 132-0031
[2]Division of Systems Engineering, Boston University, Boston, MA 02446

BOSTON UNIVERSITY

Rafik B. Hariri Institute for Computing and Computational Science & Engineering

Introduction: Project and IL

In imitation learning (IL), an agent learns from "demonstrations" conducted by an "expert". IL was used to develop a navigation policy leveraging data collected from rodents during foraging. Key components of the IL process include state representation, actions, and a reward function. A neural network was trained to map states to actions

<u>Expert Demonstration:</u>

Demonstration $D = \{\tau_1, \tau_2, \dots \tau_m\}$ given transition τ based on state s, action a (a 2d vector with x and y components), and index i

<u>Training a policy π (modeled by a NN)</u>

$$\pi: s \rightarrow a$$

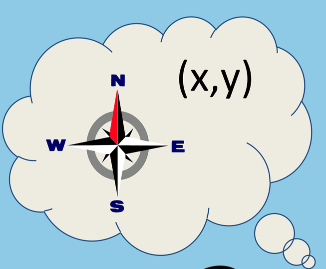
$$NN: S \rightarrow (\mu, \Sigma)$$

$$a \sim \mathcal{N}(\mu, \Sigma)$$

$\tau = (s_i, a_i)$
$i = (1,2 \dots m)$
Data: $a_i = [a_{x,i}, a_{y,i}]$

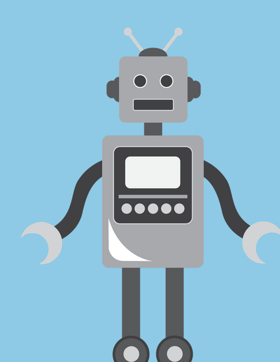
<u>Minimizing Loss through bivariate normal distribution:</u>

Where p is the correlation between x and y, μ is the location parameter vector, and Σ is the covariance matrix:



$$\Sigma = \begin{pmatrix} \sigma^2{}_x & \rho\sigma_x\sigma_y \\ \rho\sigma_x\sigma_y & \sigma^2{}_y \end{pmatrix} \qquad \mu = \begin{pmatrix} \mu_x \\ \mu_y \end{pmatrix}$$

$$-\log\left[\det(2\pi\Sigma)^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(a_i - \mu_i)^T \Sigma^{-1} (a_i - \mu_i)\right)\right]$$

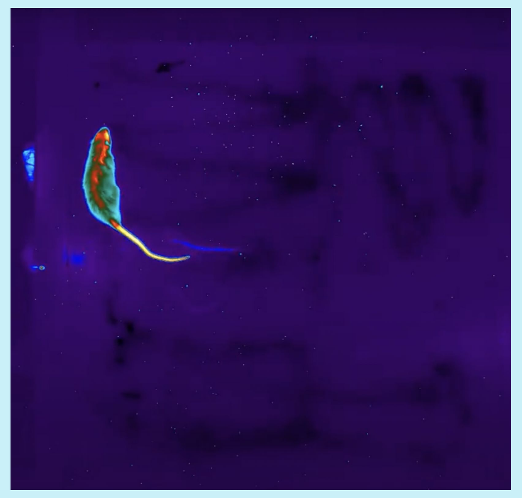


Expert

MDP (Markov Decision Process) Environment

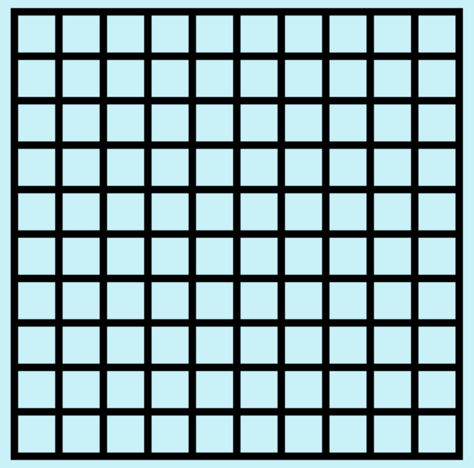
Agent

Data Processing



Midpoints taken and recorded over 125172 frames into CSV file

Discretized to a 10x10 gridworld



- Two locations of food source (3,1) and (4,9)
- Location of food determined by most frequent visiting points
- Sequence of consecutive 2D points used to consider distance from agent to both food sources ⟶ generate states (4D vector) and actions (2D vector)

Methodology

Policy 1: Deterministic

- Takes in one input that corresponds to one output
- Two kinds used, one with 3 inputs the other with 5

Policy 2: Stochastic

- For one input outputs a *probability distribution* of actions
- Better representation of real-life scenario

Environment:

State:

$$s_t = [x_t, y_t, d_{1,t}, d_{2,t}]$$

Action:

$$a_t = [a_{x,t}, a_{y,t}]$$

Reward:

$$R = (1\text{-}\ d)\ if\ d \le 5$$
$$R = 0\ if\ d > 5$$

Transition: t: (s,a) → s'

$$t: (x_t, y_t, d_{1,t}, d_{2,t}, a_{x,t}, a_{y,t}) \rightarrow (x_{t+1}, y_{t+1}, d_{1,t+1}, d_{2,t+1}, a_{x,t+1}, a_{y,t+1})$$

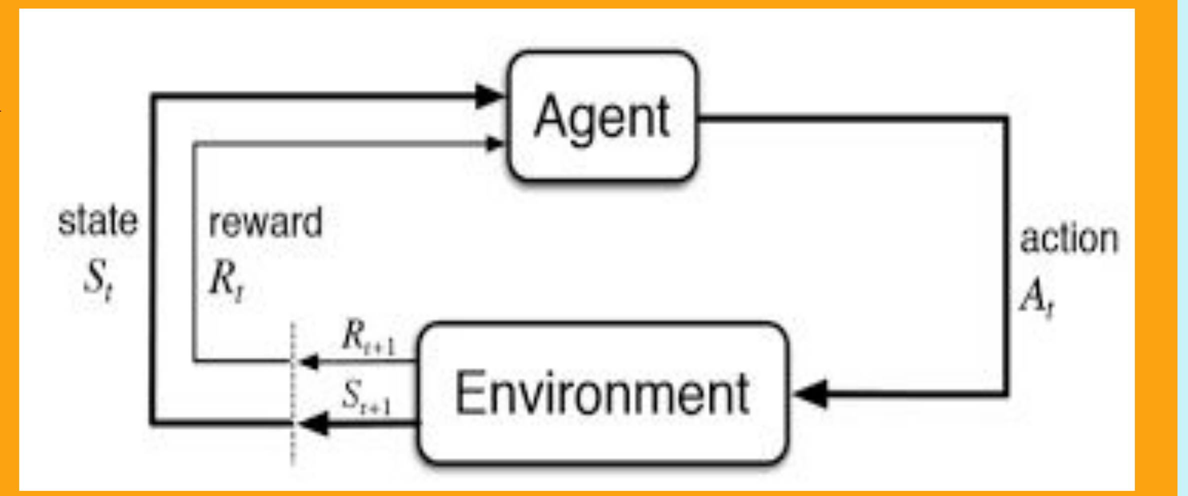
$$x_{t+1} = x_t + a_{x,t} \qquad y_{t+1} = y_t + a_{y,t}$$

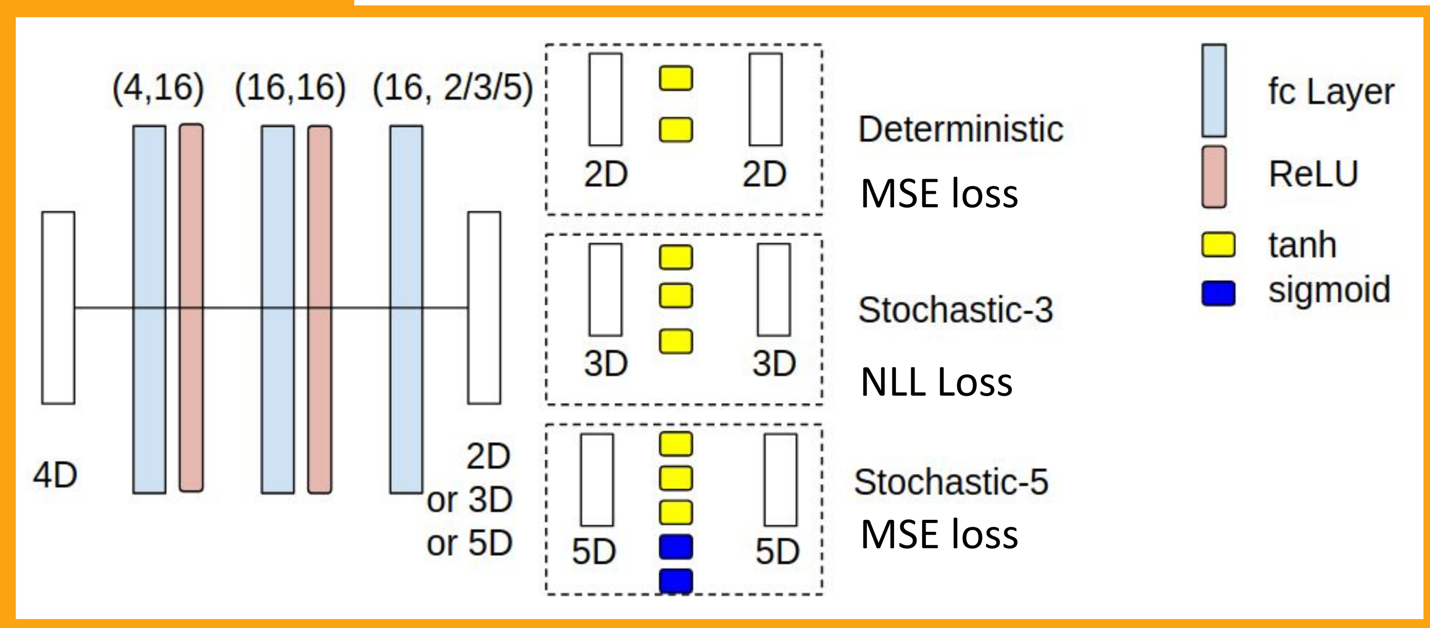
$$d_{1,t+1} = \sqrt{(food_{1,x} - (x_t + a_{x,t}))^2 + (food_{1,y} - (y_t + a_{y,t}))^2}$$

$$d_{2,t+1} = \sqrt{(food_{2,x} - (x_t + a_{x,t}))^2 + (food_{2,y} - (y_t + a_{y,t}))^2}$$

- s_t as state at time t
- x_t as x coordinate at time t
- y_t as y coordinate at time t
- $d_{1,t}$ as distance from rodent to food 1
- $d_{2,t}$ as distance from rodent to food 2
- R as reward
- a_t as action at time t
- $a_{x,t}$ as direction and length traveled by the agent along the x-axis
- $a_{y,t}$ as direction and length traveled by the agent along the y-axis.

Figure 1: Transition Process





Results and Analysis

Action	(-1,-1)	(-1,0)	(-1,1)	(0,-1)	(0,0)	(0,1)	(1,-1)	(1,0)	(1,1)	Total
Frequency	11	471	12	1095	122006	1084	3	481	9	125172
Percentage.	0.009	0.376	0.010	0.875	97.471	0.866	0.002	0.384	0.007	100

Table 1. Statistical Analysis of Action Value in the Data

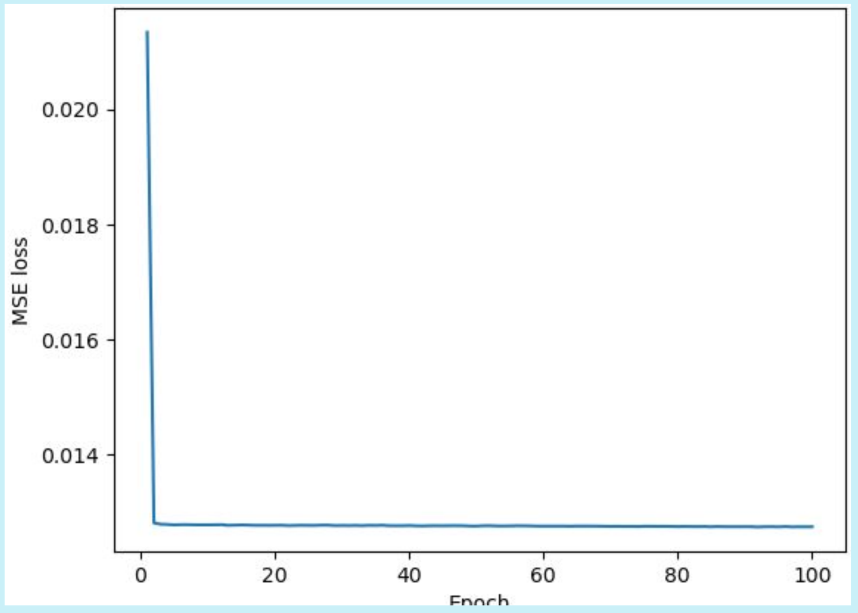


Figure 2. MSE Loss Curve of Deterministic Policy Training

- Coordinate difference of two consecutive points calculated (integer) and distances collectected
- (0,0) Majority
- Deterministic policy quickly converges
- 5 Parameter stochastic policy yielded lower NLL loss
- Mean cumulative reward calculated (1000 transitions)
- Action distribution visualized through heat map (ft. both goal points)
- Heat map in support of method

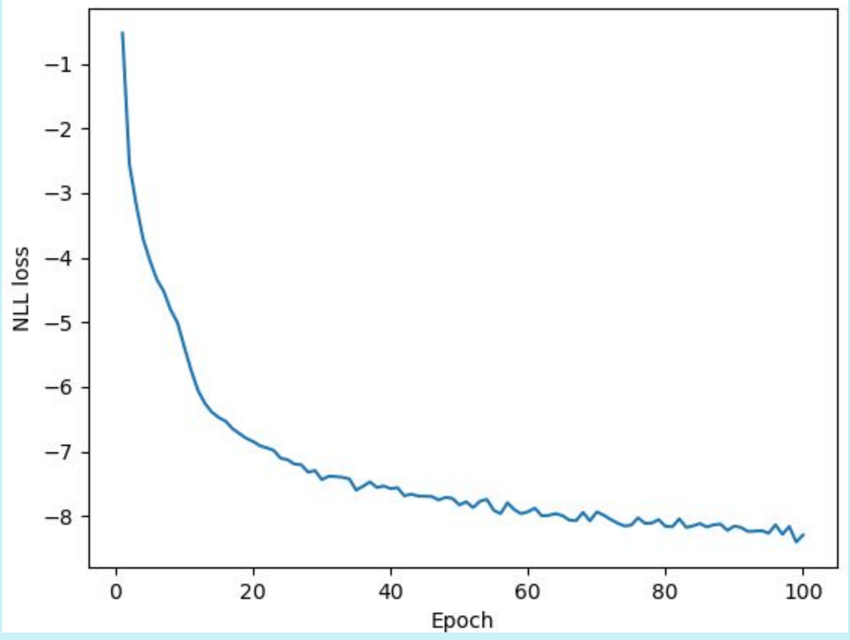


Figure 3. Bivariate NLL Loss Curve of Stochastic Policy (3D output) Training

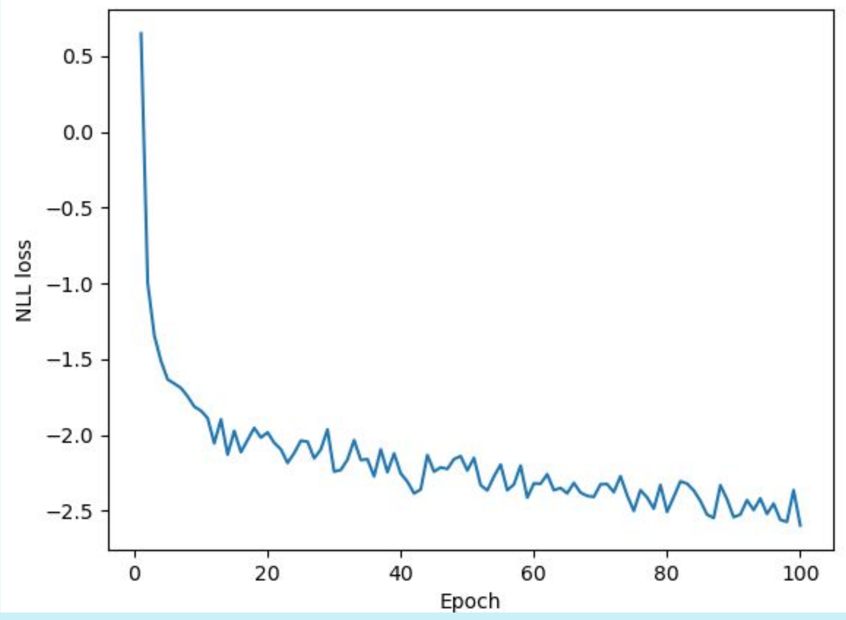


Figure 4. Bivariate NLL Loss Curve of Stochastic Policy (5D Output) Training

	Deterministic	Stochastic-3	Stochastic-5	Random Policy
Cumulative Reward	0	109.3	192.6	145.5

Table 2. Cumulative Reward Results of Different Policies Obtained from 10 Trajectories

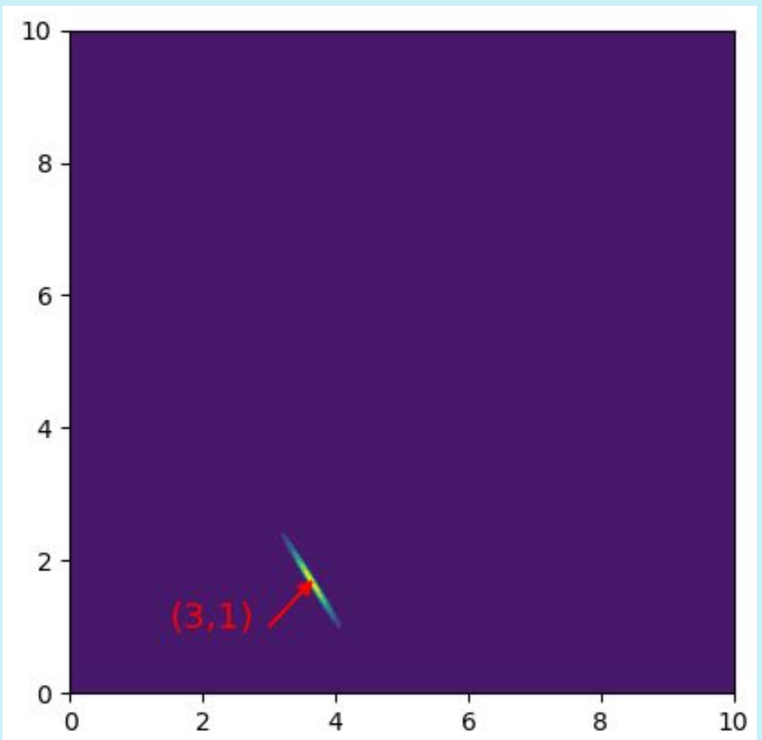


Figure 4. Visualization of Action Distribution in Point (3,1)

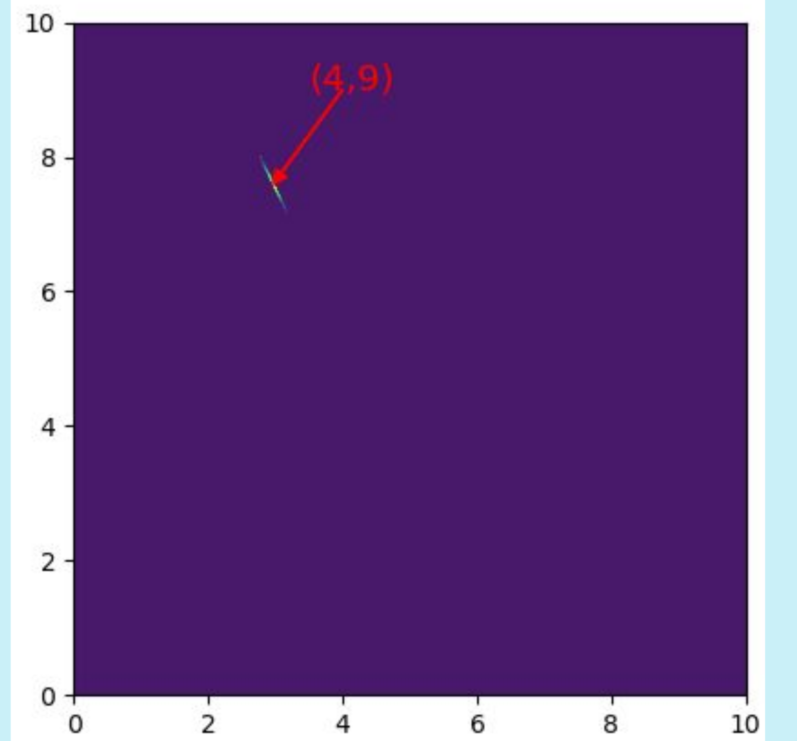


Figure 5. Visualization of Action Distribution in Point (4,9)

Conclusion

This investigation presents a method to learn navigation from real rodents through imitation learning. Initial experimental results show the effectiveness of the proposed method in comparison to the random policy. Visualization of the action distribution also provides evidence in support of the method.

Potential Future Improvements:
A possible future research direction could be to combine the learned navigation policy with locomotion policy to achieve higher autonomy. Another direction is to explore the adaptability of the learned policy on other different tasks.

References and Acknowledgements

[1] Giammarino, V., Dunne, M. F., Moore, K. N., Hasselmo, M. E., Stern, C. E., & Paschalidis, I. C. (2022). Learning from humans: combining imitation and deep reinforcement learning to accomplish human-level performance on a virtual foraging task. *arXiv preprint arXiv:2203.06250*.
[2] Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.
[3] et al., 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library. In Advances in Neural Information Processing Systems 32. Curran Associates, Inc., pp.

I'd like to thank Xuelei Chen, Vittorio Giammarino, and Professor Paschalidis for their assistance for this project. Additionally, I'd like to thank BU RISE for providing this research opportunity.