CS698R: Deep Reinforcement Learning Project

Final Project Report #5

Team Name: DarkBox
Project Title: Foraging in the Field
Final Project #: 5
Team Member Names: Roll No
Prakhar Srivastava: 170486
Tabish Ahmad: 21111060
Mohammed Niyas P: 170394
Areeb Ahmad: 180135
Samarth Mehrotra: 20128408

1 Introduction

Our problem concerns with the basic behaviour of human beings or animals on foraging where we have to maximize the benefits and minimize the costs. Animals have to decide when they are foraging in a patch, when is the best time to leave and move on to another patch. It emphasizes on the strategic nature of a risky choice of leaving the patch in long term. By using Deep Reinforcement Learning we will try to understand how to forage optimally in a field with patches of berries.

Significance in Neuroscience:-
Foraging behaviour is highly studied topic in neuro-science especially in the branch of behavioral ecology. Foraging theory deals with observing and explaining foraging behaviours of species in response to the environment it lives in. This problem helps in studying decision making of species to continue their journey in the hope of finding new patch of resources given the cost of travelling.

Significance in Reinforcement Learning:-
From reinforcement learning perspective, it is an optimization problem. But the design is such that you will end up with rewards less than that you started with. Its about saving as much as you can. Also, given the size of environment and time limit, computational efficiency is crucial.

The final objective of the project is to present an agent which can forage optimally in the given environment.

2 Related Work

Autonomous Foraging with SARSA-based Deep Reinforcement Learning : - This paper suggests SARSA(State Action Reward State Action), neural networks and computer vision for foraging tasks. The environment is lot more complex than this problem and is focused on distinguishing poison from food. The equation for SARSA is:

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

where α is the learning rate and the next action a' is given by the current policy.

Marginal Value Theorem : - Proposed by Eric Charnov in 1976, it describes the behavior of an optimally foraging individual in an environment where resources (berries) are located in discrete patches separated by areas with no resources.
Observation of Animals:- Species like great tits and screaming hairy armadillos have been studied and their foraging patterns have been tested for optimality.

Information foraging :- Application of optimal foraging theory to understand human behaviour while searching for information on internet.

3 Problem Statement

Aim:- Collect berries throughout the field in a way it maximizes the reward in a limited time, where berries are distributed in patches throughout the field.

- Maximize the cumulative reward at the end.
- Maximize the remaining health at the end.

The agent should be able to learn to leave the patch at optimal time and explore the environment optimally like a human should.

4 Environment Details and Implementation

- **Environment:-**
  It is essentially a berry collecting game. You have to collect as many berries as possible in 5 minutes. The graphical interface is made with the help of Pygame which is integrated with OpenAI gym.

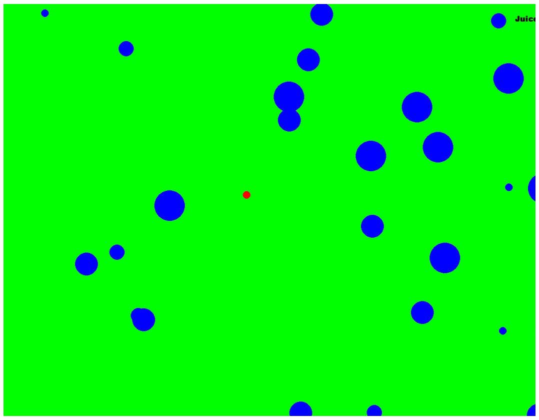


Figure 1: Environment

- **State Space**
  Set of coordinates within the boundary(20000x20000) where berry locations have rewards according to their radii.
- **Observation Space**
  $< Screen(1920 \times 1080), Depletionrate(timet), RewardsAccumulated(timet), AgentCoordinates(x,y) >$
- **Action Space**
  Remain in same cell + 8 directions - $< N,S,E,W,NE,NW,SE,SW >$
- **Implementation:-**
  Used opengym and pygame. Pygame provides the functions required to build the GUI for berry collecting game. It is integrated with gym environment so that agent can also play and algorithms can be applied.

5 Proposed Solution

The agent is trained using the algorithm Double Deep-Q Learning(DDQN). The agent's state size is 35 and action size is 8 or 9 depending if it has the option to stay.

The features can be divided into 4 vectors of length 8. Each vectors denotes the direction: [N, S, E, W, NE, NW, SE, SW]. The agent scans in each of the contained direction and if it encounters a berry of size say 'n' it fills the corresponding indices with $\frac{size}{distance}$ for the vector corresponding to size 'n'. Therefore we have 8*4 = 32 states.

The 33^{rd} state is **density** which tells how much of the visible screen is occupied by the berries. The 34^{th} and 35^{th} states are the remaining health and remaining time respectively. The features are scaled to give the best results.

Added noise to the states so as to prevent over-fitting. $[states] + \epsilon$ where $\epsilon \sim N(0, \sigma^2)$. To reduce the stuttering motion of the agent we don't take action from the agent's q-network every time. Instead, the action is taken from the q-network with a ϵ_2 probability.

The greedy policy is updated so that the random action is continued for n seconds, since the environment is so large a single-pixel action wouldn't give any information.

6 Results and Analysis

Berries reward according to size is taken as [4, 3, 2, 1]. Our Model took 26hrs to train without GPU.

Social Experiment:

Volunteers were tasked to play the foraging game. Data of over 183 human players was collected.

Given below (Fig.6) is the frequency distribution of human performance. The horizontal axis shows the total reward, and the vertical axis indicates the number of people collecting that much reward.

The mean(μ) of human performance is 344.7978 and standard deviation(σ) is 114.75,with the highest score of 600.

Similarly we ran the agent(After training on 280 episodes),for 50 times.The results were outperforming the humans,mean(σ) was 416.528 ,and standard deviation of 176.The highest score was 750.

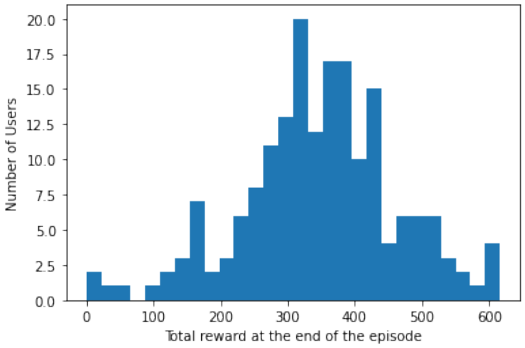


Figure 2: Human Performance

Running Average Reward vs episodes:

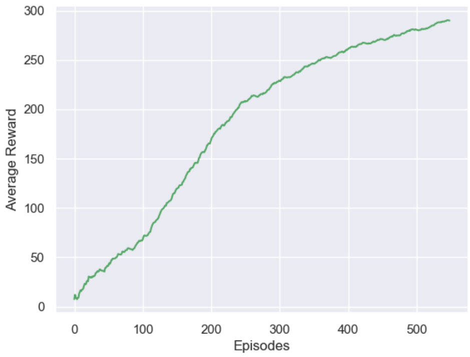


Figure 3: Average reward vs Episodes

The agent performs well after reaching a cumulative score of around 250 and completes the given objective of the game. The agent can be further trained for 500 episodes to reach total convergence.

Health vs Time

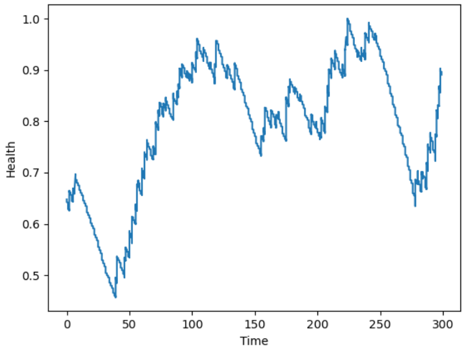


Figure 4: Health vs Time

The health vs time graph shows us when the agent decides to leave the patch and can also tell us about the amount of patches visited.
Agent's path

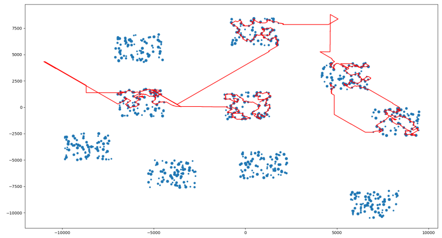
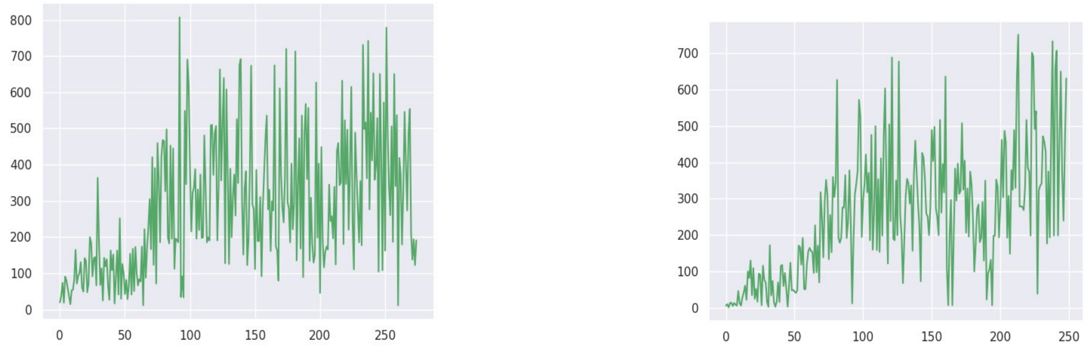


Figure 5: Agent's path



(a) Non-continuous motion(agent can choose not to take any action)

(b) Continous motion

Figure 6: Total reward vs episodes

7 Experiments

In one of the training sessions, the agent reached an unbelievable score of 1297.It can be treated as an outlier .Given below is the path traced by the agent in that episode.

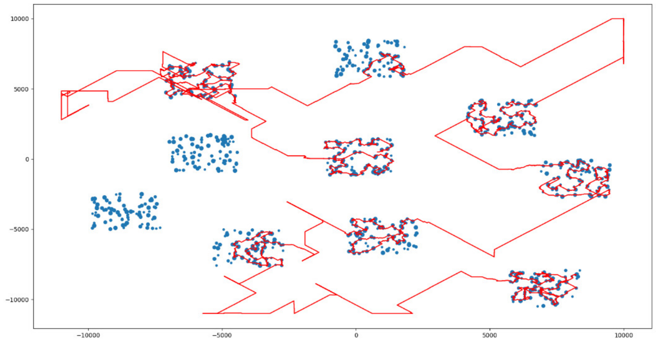


Figure 7: Outlier perfomance

We also experimented with NEAT and actor-critic methods but that requires higher computing power and resources to work properly. DQN and Dueling-DQN were working more or less the same.

8 Future Directions

A CNN-based approach can be implemented to learn more complicated features. Further optimizations will need to be made while using CNN since saving the screenshot of the image while the game is running drops the FPS drastically.To deal with that, we can try cropping the region around the agent(say 210 × 120 pixels) and will use it as a CNN input, and it may also mitigate the oscillation of the agent. Furthermore, a GPU will be needed to train the CNN-model efficiently.

Actor-critic methods can be tested to get low variance during training times. As we have seen from the plots the DQN and DDQN gives a lot of variance while training the agent.

A genetic algorithm NEAT(Neural Evolution of Augmented Topologies) can also be used to get a optimal agent after some generations.

9 Conclusion:

We demonstrated the use of reinforcement learning techniques for a game aimed to study the foraging behavior of humans. Our trained agent was able to match with the human accuracy and play with human-like features. The agent can still be further optimized by removing the stuttering nature of the agent and having a more robust exploration strategy. Advances methods of reinforcement learning like PPO can be expected to lower down the variance during the training making the agent more stable. Our model was also lightweight with inference time also very less which causes no delay in the movements of the agent.

10 Member Contributions

Name	Contribution (Milestone-1)	Contribution (Milestone-2)
Prakhar Srivastava	Environment,Pygame ,slides	Environment, Pygame and model implementation, Literature Survey, Solution approach, Feature selection, Training agent, getting results, final slides and final report.
Tabish Ahmad	Literature Survey, Environment,slides	Environment, Feature selection, Literature survey.
Areeb Ahmad	Environment, Solution Approach,slides	Environment, Solution Approach, Training agent, plotting results, final slides and final report, Data Analysis.
M.Niyas P	Pygame, Literature Survey,slides	-
S. Mehrotra	-	-

References

Cassini, Marcelo H., Alejandro Kacelnik, and Enrique T. Segura. The tale of the screaming hairy armadillo, the guinea pig, and the marginal value theorem. *Animal Behavior*, 39(6):1030–1050, 1990.

Eric L. Charnov. Optimal foraging: the marginal value theorem. *Theoretical Population Biology*, 9:129–136, 1976.

R. J. Cowie. Optimal foraging in great tits (parus major). *Nature*, 268:137–1393, 1977.

Risto Miikkulainen Kenneth O. Stanley. Evolving neural networks through augmenting topologies.

Anderson Mesquita, Yuri Nogueira, Creto Vidal, Joaquim Cavalcante-Neto, and Paulo Serafim. Autonomous foraging with sarsa-based deep reinforcement learning. In *2020 22nd Symposium on Virtual and Augmented Reality (SVR)*, pages 425–433, 2020. doi: 10.1109/SVR51698.2020.00070.

Tony Russell-Rose and Tyler Tate. Designing the search experience : the information architecture of discovery (chapter2: information seeking). 12 2012.

Davi D W. Stephens. Decision ecology: Foraging and the ecology of animal decision making.