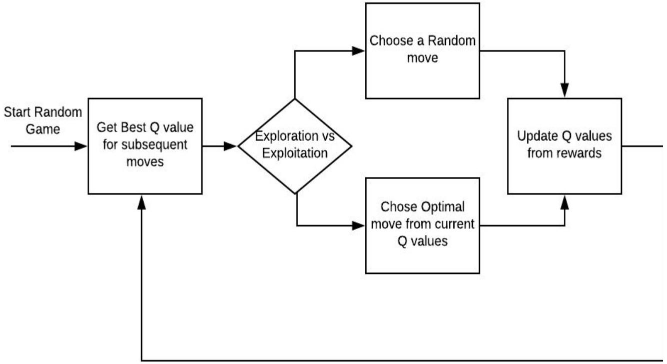
Connect Four Bot using Reinforcement Learning

Team Members:
Anupama Anupama
Harshini Muthukrishnan
Prashil Bhimani
Santosh ALN

Modelling the Problem

- We are modelling the problem using reinforcement learning and will use Q learning to learn a policy to play the game.
- Is a popular reinforcement learning algorithm.
- Good moves that lead to rewards.
- Not feasible to use Q Table to keep a track of values of all states and actions because of large number of values
- So instead we will train a neural network to give us Q values to decide the next best move
- Every hole in the Connect 4 will be represented as
  - 0 - Empty and not reachable
  - 1 - Filled by player 1
  - 2 - Filled by player 3
  - 3 - Empty but reachable in next move


Start Random Game
Get Best Q value for subsequent moves
Exploration vs Exploitation
Choose a Random move
Chose Optimal move from current Q values
Update Q values from rewards

Q Learning Algorithm

Q Learning

- Q Learning value update equation is :
  - V new(st) ← V (st) + α(rt + γV (st+1) – V (st))
- But neural network has its learning rate and hence we set α=0
  - V new(st) ← rt + γV (st+1)
- Important parameters:
  - ε - Exploration rate - Initially the algorithm must explore more and as the algorithm proceeds we have to reduce this value
  - γ - Discount factor - For every game as the time passes we can give more importance for short term rewards rather than long term rewards.

Neural Network

- A fully will not be as a good strategy to approximate such a vast search space hence we will use structured neural network instead.
- Neighbours of a block can influence its importance
  - Eg: A red block surround by all blue is not useful but if it is surrounded by red it will be very useful
  - 3 in a row on a +/-ve diagonal, vertical and horizontal rows are important
- Will use 1-hot encoding to encode the categorical input(0-4)
- α (learning rate of neural network) will be decrease over time

Work Done:

- Developed the game framework to play against different players
- Developed min max player to play against the model
- Developed a basic structure for the neural network

Need help in:

- Developing and selecting different parameters for neural network
- Verifying and understanding the effects of parameters in neural network.
- Making the algorithm better to understand overfitting
- Determining whether rewards for win should be same always for all wins or faster wins will mean more rewards?

Thank You