Project: Best blackjack player in the world
Teammates: Haoyu Wu, Raphael Liu.
NetId: hwu36, raph651

1. The main project idea

We want to simulate the gambling game “Blackjack” in a multiplayer setting. The game is modeled as two agents. The first agent corresponds to a player who wants to maximize profits. The second agent corresponds to the dealer who wants to minimize the profits from all the players.

For the player, we expect to build a deep neural network $f_\theta(s,a)$ with parameters θ, state *s* and action *a*. This neural network will approximate the given (s,a) and return a (p, v) pairs, where p is the probability for each action, and v is the value.

We may first assume the dealer agent is fixed and his action is hard-programmed: the dealer will always hit and get another card if the current sum is less than 17, will stand otherwise. If time allowed, we expect to build a separate neural network that takes the current state as input and outputs the same (p,v) pairs from the dealer perspective.

The state s for this model will be the sum of the current cards in hand, 1 revealed card in the dealer, and most importantly, the number of each card in the bankpool. For the player, the action a is defined as three options: hit, double, or stand. For the dealer, the action a’ is defined as two options: hit or stand. The transition model in this game is that the player gets a random card from the bankpool so a new state s’ will be determined.

The goal of this project is to combine the neural network and Monte Carlo tree search to find the optimal strategy for the player and dealer, separately.

2. Some possible related work you would refer to
Mastering the game of Go without human knowledge (Neural network with Monte Carlo Tree Search).

3. A proposed method (or several)
We may first assume that the dealer’s strategy is fixed: stand for 16 or less and hit for 17 or more. Later stage if time allow we will explore dealer’s strategy with neural nets (so it’s kind of like GAN, players want to maximize profit while dealers want to minimize players’ profit)

There are two problems to solve for players: 1) how much to bet. 2) what actions to make given certain hands. The reason for the separation is that the bet is important to maximize our profit but our actions for each round does not depend on our bet amount.

We can design a supervised model (NN, regression…) that takes the remaining deck as input and outputs the bet amount. The return of the round will be feed into some procedure that determines how to backpropagate (e.g. If the player wins 100, we can generate a label to be 100*1.1=110 and use 110 as the real label to back propagate; if player lost 1000, we will punish the risky bets by generating a label 1000*0.8. We haven't figured out an ideal way to do this yet).

For the second problem, we first define the MDP problem. Our state space is an array of 15 elements (for 2 players, 14+n elements for n players). 13 elements store the remaining count of the cards for 2, 3,..., king, ace; 2 elements store the sum of the hands for players and dealers. Actions are [Hit, Double, Stand]. Transition model is a stochastic transition that simulates a card being dealt to players (for Double and Hit actions) or make the current state as the end of state (for Stand action). Reward is 1 if the player won or 0 otherwise.

Then we design a neural net $f_\theta(s,a)$ that takes our state and action as input and outputs an estimated value for the next state x'. Exploration-exploitation tradeoff will be considered to choose our actions. At the end of the round, we use the reward to update the value estimate for each state we end up in (maybe using n-step TD or monte carlo? Haven't decided) and use the updated value to back propagate the neural net.

4. What you expect the outcome to be

We expect the outcome to be the solutions to our two problems at a given state: 1) how much to bet 2) what actions to make. The first solution gives us the final supervised model. The second gives us the neural network that leads the player to make the best action.

5. Potential risks you see in the project

Many model to train, could be hard; Catastrophic interference; Performance of the neural network might depend on configurations (how much layer, layer size, activation functions, etc.);

6. Questions to be addressed:

- What are some impacts of this research?

  Our model serves as a good example that an AI model could learn and adapt from interacting with an unknown environment and achieve good results without defining rules. Such methods could potentially have further applications into fields such as financial securities, autonomous driving, etc. It allows our model to adapt to a changing environment.

- What is novel about the approach you are taking?

  We use the online learning approach and let the Blackjack agent and Dealer agent play in an adversary way. We start from 0 dataset and update the neural networks to obtain optimal strategy without expert knowledge in the field. Our research involves a non-deterministic transition model that is unlike the AlphaGo model, where a new action changes the state in a deterministic way. Our research separately trains different models to optimize the combined strategy for betting and playing during a whole game. This combined setting will be powerful in simulating more complex real-world games or models. The game rules can be easily changed and adapted in a manner that doesn't affect the overall structure.

- How do learning and/or probabilistic inference techniques play a key role?

Learning and prob inference are used to construct, update value function and choose actions and bets.

- What is your metric for success?

  Money money money money as much as possible! Rate of profit will be compared with historical data.

- What are key technical issues you will have to confront? Are there any other big challenges?

  Tuning NN structures and trying different online learning parameters is time and computationally costly.

- What software or datasets will you use?

  Our model trains itself. Currently no software is considered needed.

- What is your timeline? Include specific targets for the progress report.

  11/14: building the big structure of the learning environment; implementing the game process and dealers actions.

  11/21: implementing the NN for bets and value function. Implementing the training process.

11/28: Tuning and training models.

12/05: Presentations