Project Report

Cribbage AI

COMP 4106 Artificial Intelligence

Adam Prins

100 879 683

**Introduction**

Cribbage is a two-player card game with a semi-complicated scoring system, and asymmetrical gameplay. A turn has three parts, the crib, the play, and the count. The winner is the first player to score more than 120 points.

Cribbage requires players to decide what 4 cards to keep from the 6 they are dealt, while considering what card might be turned up, and its effects on the cards they keep. Then the players must decide how best to play their cards to capitalize on points, while trying to minimize how many points their opponent earns.

The objective of the project is to create an artificial intelligence that is able to play the game Cribbage. It must be able to select what cards to keep, and which to put in the kitty. It must be able to determine how to play its cards in order to score points, while trying to prevent the opponent from doing so.

The source code can be found at:

<https://github.com/AdamPrins/CribbagePlayer>

**Motivation**

I wanted to do this project in order to better understand the thought process and decision making that goes on during a game of Cribbage. There is a lot of strategy and consideration that goes on during a game, and not all of it was something that I had explicitly thought about and considered before.

I wanted to work on this game in particular because of its complicated scoring system and asymmetrical gameplay. Where the Dealer gets points from a second hand of cards, the kitty, while the opponent gets to score their hand before the dealer.

**Methods**

The Agent was designed as a utility-based Agent, who’s utility function is the number of points earned by engaging in certain behaviours. The Agent determines the possible remaining cards in the deck, using the known cards it drew. The agent then calculates the likely points that the hand will score, and the points that go into the kitty, for each possible hand it keeps. There are 15 such possible hands it can create from the 6 cards the Agent is dealt. The points that go into the kitty are considered positive for the dealer, who scores the kitty, and negative for the opponent, who does not. The hand that is likely to give the most points is the one kept.

For the play, the agent evaluates which of its cards earns the most points and plays it.

**Results**

The AI is able to determine what hand is most likely to net them the most points. This decision also includes the points going into the kitty, as the dealer will earn points from it, they want to put points into it, and the opponent will want to avoid putting points into it.

The AI can

**Implications**

Through this project, I have come to the conclusion that the strategical element of Cribbage is more complicated that I had originally given it credit. I am quite happy with the way it determines which hands to keep, but the AI for the play is shallow. The play involves more planning and manipulation of your opponent than I had thought, and is not a simpler determination of what card gives you the most points.

**Future Work**

The AI has places where improvements can be made to improve its performance.

The current AI does not consider how many points might be earned from playing cards when selecting a hand to keep, only what hands will give the most points afterwards. This can be very important for play near the end of the game. The dealer always counts hand points second, so if both players are close to winning, the dealer can prioritize trying to peg points, so that they can win before the opponent can count the points in their hand.

Another point of improvement is the play itself. Right now the AI only looks to the immediate rewards to determine what cards to play. Improvements can be made so that it also actively tries to avoid setting up points for the opponent. And an improvement to that would be plays that try to lead an opponent into setting you up, like setting up a double when you know you can play a triple, or a shorter run, that you could make longer.

**User Manual**

The project is built using python 3, and has a dependency on the package *termcolor*. In order to run the code, install the dependency for python 3. The common way to install dependancies is pip, and the command would look like this: pip install termcolor

The test cases can be run from test.py. If all the tests pass, a message will return saying such, but if some of the cases fail, then an assertion error will be returned signifying which part of the code failed the test. The AI can be made to play a game against itself by calling cribbage.py. A log of the game will be output into the terminal, showing the round, cards dealt, how cards are discarded, how the hands are played, and all the scoring for the round. When one of the players reaches 121 points or more, the game will end, and a message saying who one will be displayed.

The source code can be found at:

<https://github.com/AdamPrins/CribbagePlayer>