

2018年度 卒業論文

Pythonによるテキサスホールデムポーカーの役判定と勝敗決定

龍谷大学 理工学部 数理情報学科

t150007 伊丹 彰希

担当教員 佐野 彰

概要

一昔前の日本では、5枚の手札で数回交換し、役を作るというドロポーカーが主流であった。実際に、私たちが子供の頃に遊んでいたポーカーもドロポーカーが多かったのではないかと思う。2018年にカジノ法案が可決し、近々日本にもカジノが設立するかもしれない。そのこともあり、近年の日本でもドロポーカーからテキサスホールデムポーカーに主流が変化しつつある。本研究では、テキサスホールデムポーカーを理解し、活用するためにポーカーゲームをPythonを用いて作成した。

本研究で作成したポーカーゲーム上のトランプやプレイヤーの手札は、4種類のマークとAからKまでの13枚の数字に対応させた 4×13 の2次元配列で表現した。また、配列は、ゲームの進行に伴って常に更新される。

カードを配る機能では、関数を1つにせず、各プレイヤーに配る関数とコミュニティカードと呼ばれる場で全プレイヤーに共用されるカードをオープンする関数に分けることでそれぞれのカード配布を利用しやすくした。役による強弱判定は、役による強さを数値化することによって勝敗を決定しやすくなっている。また、同じ役でも、役を構成するカードの数字によって強さが異なっているので、手札が全く同じでない限りカードの数字によって強弱がつけられるようにした。さらに、役によって強さを表す数値の範囲が異なるように数値化のアルゴリズムを考え、役による強弱の逆転が生じないようにした。

プレイヤーの勝利判定では、上記の役による判定のみで勝敗が決する場合は問題ないが、1ペア、2ペア、3カードなど全く同じ役を持つときは、残りの手札で勝敗を決定する必要がある。したがって、それぞれのプレイヤーのハイカード(役以外の最も強いカード)を比較する工夫を施した。ポーカーにおいて1番強いカードがAであるので数字としては1であるが、Kの13より強いので強さを逆転させなければいけない。したがって、ハイカードの判定では、最初にAがリストに入っているかチェックし、もしリスト内にAがあればAを14や140に変化させている。

役の成立をチェックするそれぞれの関数では、1番強い役を選択するようにした。また、役が成立していなくても空のリストと役なしの数値化を表す値0を返すようにしている。各プレイヤー毎に役の成立をチェックしていき、役が成立していれば、その時の役と数値化された値をそのプレイヤーに記憶させている。したがって、強い役が成立するたびにそのプレイヤーの役と値は更新されていく。

本論では、最後に上記の研究内容と作成したプログラムを客観的に考察し、改良点、工夫した点などを記述している。