

Assignment 5

Card counting

|  |  |  |
| --- | --- | --- |
| 과목명 |  | 확률및통계 |
| 담당교수 |  | 심동규 교수님 |
| 학과 |  | 컴퓨터정보공학부 |
| 학년 |  | 3학년 |
| 학번 |  | 2019202009 |
| 이름 |  | 서여지 |
| 제출일 |  | 21.05.23(일) |



1. 과제설명

이번 과제는 카드카운팅을 적용하여 블랙잭 게임에서 승리할 확률을 높이는 과제이다. 카드카운팅은 게임에서 숫자가 공개된 카드의 수를 모두 기억하여 다음에 나올 카드를 추측하는 것에 이용하는 것이다. 딜러와 플레이어1, 2가 블랙잭 게임을 진행한다. 플레이어 1은 카드카운팅을 이용한 방법으로 hit, stand를 선택하고, 플레이어2는 딜러 룰에 따라 hit또는 stand한다. 게임을 여러 번 진행한 뒤, 플레이어1과 2의 승리, 무승부, 패배 횟수를 각각 출력한다.

1. 접근방법
   1. 카드카운팅을 어떻게 적용했는지 설명

플레이어1은 게임 중 카드에 적힌 수가 공개될 때마다 그것을 기록한다. 그리고 자신의 차례가 되었을 때, 기록한 카드의 수를 이용하여 특정 카드가 나올 확률을 계산한다.

플레이어1은 가장 먼저 딜러의 카드 합을 계산한다. 딜러는 16이하의 카드를 가질 경우 항상 hit 하므로, 딜러의 카드 합은 17, 18, 19, 20, 21, bust의 6가지 경우의 수를 갖는다. 모든 경우의 수에 대해 확률을 계산하여 float 배열(dealerArr)에 저장한다. 그 다음, 자신이 hit했을 경우 다음에 뽑게 될 카드의 확률을 계산한다. A에서 K까지 13개의 경우의 수에 대한 확률을 계산하여 또다른 float 배열(nextCardArr)에 저장한다.

플레이어1은 두 float 배열의 내용을 이용하여 hit하는 것이 유리한지, stand하는 것이 유리한지 계산한다. 예를 들어 현재 자신의 카드의 합이 12인 경우, 딜러의 카드 합을 17로 가정하면 플레이어1이 hit하여 9이하의 수를 뽑아도 패배하지 않는다. 플레이어1은 dealerArr에 기록된 ‘딜러의 카드 합이 17일 확률’과 nextCardArr에 기록된 ‘9 이하의 카드가 나올 확률’을 곱연산하여 hit가 실패하지 않을 확률(pHit)에 기록한다. 6가지 딜러 카드합의 경우의 수에 대해 모두 반복하여 hit가 실패하지 않을 총 확률을 구한다.

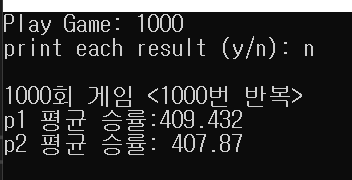
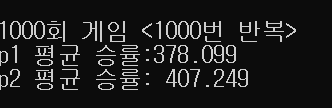
반면 stand하여 실패하지 않을 확률은 pStand에 기록한다. 예를 들어 현재 자신의 카드 합이 18인 경우, 딜러 카드의 합이 17, 18일 때 실패하지 않는다. 따라 pStand에는 ‘딜러 카드의 합이 17 혹은 18일 확률’을 기록하게 된다.

구한 pHit와 pStand의 값을 비교하여 실패하지 않을 확률이 더 큰 것을 선택한다.

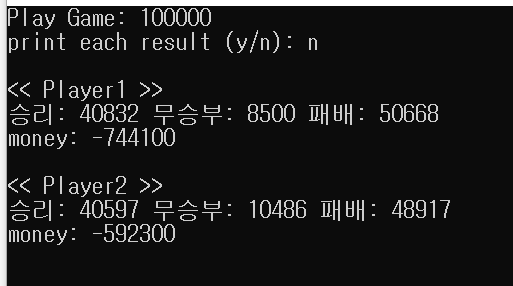
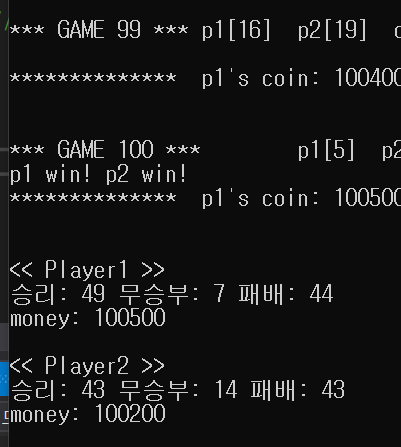
* 1. 과제를 진행하면서 생겼던 이슈는 어떻게 해결하였는지 설명

위에서 설명한 카드카운팅을 이용하는 과정에서, hit또는 stand가 실패하지 않을 확률(승리, 무승부)이 아닌, 성공할 확률(승리)을 이용하여 코드를 작성했었다. 이렇게 작성할 경우 플레이어1이 플레이어2보다 승리한 수는 더 많았지만, 오히려 보유한 돈이 더 적게 나오는 결과가 발생하였다. 플레이어1이 플레이어2보다 비긴 수가 적고, 패배한 수가 더 많기 때문에 발생하는 문제였다. 따라서 hit 또는 stand가 실패하지 않을 확률을 이용하게 되었다.

1. 실험 및 검증 결과



[ pHit > pStand ] [ pHit > pStand && pHit > pBust ]



[ pHit > pStand ]

플레이어1과 2의 승률이 크게 차이나지 않았다. pHit와 pStand에 가중치를 적용하거나, threshold를 지정하는 방법을 시도해보았으나 적합한 수치를 찾지 못해서 큰 효과를 얻지 못하였다. hit를 선택하는 경우 bust할 확률을 pBust에 별도로 저장하여 활용하는 방법은 확률을 적절하게 활용하지 못하여 오히려 승률이 낮아지는 문제가 발생했다.

위의 10만번 시행의 결과에서, 플레이어1은 플레이어2보다 약 230회 정도 더 많은 승리를 했지만, 패배의 수가 약 1750회 더 많기 때문에 더 큰 손해를 보았다. 발견한 문제점이 개선되지 않은 것이다. 따라서 카드카운팅을 이용하여 승률을 미세하게 높일 수 있었지만, 오히려 패배 횟수가 증가했음을 알 수 있다.

1. 고찰

프로젝트를 진행하며 확률 정보 자체보다, 그것을 활용하는 방법이 더 중요하다는 것을 느끼게 되었다. 모든 카드의 수를 기억하고, 모든 경우에 대한 확률을 구할 수 있지만, 그 정보를 적절하게 활용하는 전략을 찾아내지 못했기 때문에 플레이어1의 승률은 개선되지 못하였다.

과제를 진행하며 답을 찾지 못하였지만, pHit, pStand, pBust에 대해 가중치와 threshold를 적절하게 적용할 수 있다면 승률의 개선을 기대할 수 있을 것 같다. 적절한 값을 찾기 위해서는 많은 시행을 통한 데이터와 분석이 필요할 것이다.