Exploring Equity in the Golf Skins Game

Olivia Strasburg
Department of Mathematics and Computer Science
Denison University
Granville, OH 43023
Email: strasb_o1@denison.edu

***Abstract*—In the game of golf, to compete fairly with golfers of varying skill, players use handicaps which reflect their ability level. These handicaps are also used in the golf skins game. We explore how fair player handicaps make the skins game. We first use an empirical approach and simulate games to see the disparity in wins between players. We then try an analytical model to get player probabilities. Our analysis reveals an inequity in skins even when applying player handicaps.**

I. Introduction

The goal of this paper is to use statistical analysis to determine if skins is fair when played with competitors of different skill levels, and if not, to find a way to make the game equitable. Since the databases of golf scorecards are not publicly available, it is necessary to simulate data to analyse.

A. Background

Unlike in most other sports, in golf, the player with the lowest score wins. Each course has 18 holes, and each of those holes has a par. Par is the number of strokes that a professional player is expected to take to complete the hole. If a player often makes par on holes, they will have a player handicap of zero. Player handicaps are the total number of strokes a player is expected to play over par. Better players have lower handicaps while worse players have higher ones. Player handicaps are converted to a course handicap based on the rating and slope of the course the player is competing on. In a normal golf game, the player subtracts their course handicap from their total score at the end of the 18 holes for a net score. This allows for newer players to compete with experienced ones.

However, when playing skins, handicaps do not always help balance skill disparities. In the golf skins game, each hole is an independent competition. Every hole, the players compete for a prize. If multiple players tie, no one gets the winnings. Since the results on one hole do not affect the next, players' handicaps must be applied differently. To use player handicaps in skins, one point is applied to each of the most challenging holes on the course until there are no more player handicap points to apply [1]. If the handicap is greater than 18, another point is attributed to the more difficult holes in a similar fashion. For example, if a 7-handicap player is competing against a 16-handicap player, only the second player can subtract a point from their score on the 10th ranked hole. If both players use four strokes, the 16-handicap player wins since they have three strokes after applying their handicap. Even if the second player scores five strokes, the player's will tie, and no one wins the hole. This kind of scenario occurs often in skins games, leading to higher-handicap players having an advantage. We test this hypothesis in this paper.

[1]Hole difficulty is determined by golf course officials.

B. Map of Paper

This section provides an overview of the issue with using player handicaps in the golf skins game. In Section II we explore the model we use to create data. We then employ this model in Section III to follow an empirical approach to simulate games of skins. The section continues with an analytical application of the model. In Section IV we analyse the findings from these simulations. We show that players with higher handicaps have an unfair advantage in skins. Section V concludes the paper by offering future paths of study.

II. Hardy's Model

Since no public database of golf scores is available, it is necessary to create data. To simulate accurately, we employ Hardy's model[1]. In this model, Hardy proposes that players make three types of shots: ordinary, bad, and good. An ordinary shot moves the player one stroke closer to the goal, a bad shot makes no progress towards the goal, and a good shot puts the player two strokes closer to the goal. On a par 4 hole, the fewest number of shots a player can make is two good shots since each shot would make two strokes worth of progress. As described in *The Hardy distribution for golf hole scores*, the probability of a good shot is p, the probability of a bad stroke is q, and the probability of an ordinary stroke is $1 - p - q$[2]. The p and q values depend on the player's handicap and vary based on each hole's difficulty. Below is a transition probability matrix for a par 3 hole. Since we are looking at a Markov process, what happens in a previous state does not affect the next[3].

$$\begin{array}{c|ccccc|}
\text{States} & 0 & 1 & 2 & 3 & 4 \\
\hline
0 & q & 1\text{-}p\text{-}q & \mathrm{p} & 0 & 0 \\
1 & 0 & q & 1\text{-}p\text{-}q & p & 0 \\
2 & 0 & 0 & q & 1\text{-}p\text{-}q & p \\
3 & 0 & 0 & 0 & 1 & 0 \\
4 & 0 & 0 & 0 & 0 & 1 \\
\hline
\end{array}$$

This is also represented as a state transition diagram.

We can see that although the hole is a par 3 hole, there are four states. On a par N hole there is an N+1 state to absorb

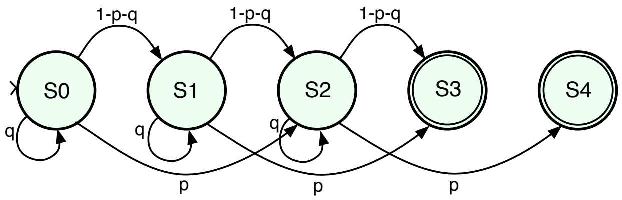

Fig. 1. State Transition Diagram of a par 3 hole

in case you make a good stroke in the N-1 state. This means that both states 3 and 4 are absorbing (stop) states.

We can use these p and q values in a number of ways. For instance, We can calculate the probability of a player taking n strokes to finish a par 3, 4, or 5 hole given any p and q [2]. The par 3 formula is given by

$$P(T_3 = n) = \binom{n-1}{n-2} q^{n-2}(p^2 + 2p(1-p-q)) + \binom{n-1}{n-3} q^{n-3}(p(1-p-q)^2 + (1-p-q)^3) \quad (1)$$

the par 4 formula is given by

$$P(T_4 = n) = \binom{n-1}{n-2} q^{n-2} + \binom{n-1}{n-3} q^{n-3}(2p^2(1-p-q) + 3p(1-p-q)^2) + \binom{n-1}{n-4} q^{n-4}(p(1-p-q)^3 + (1-p-q)^3 + (1-p-q)^4) \quad (2)$$

and the par 5 formula is given by

$$P(T_5 = n) = \binom{n-1}{n-3} q^{n-3}(p^3 + 3p^2(1-p-q)) + \binom{n-1}{n-4} q^{n-4}(3p^2(1-p-q)^2 + 4p(1-p-q)^3) + \binom{n-1}{n-5} q^{n-5}(p(1-p-q)^4 + (1-p-q)^5) \quad (3)$$

We can also determine the average number of strokes a player will take on a par m hole.

$$\mu_m = \Sigma_{j=1}^{m} - \frac{(m+1-j)p^{j-1}}{(q-1)^j}, m = 1, 2, 3, ... \quad (4)$$

these formulas allow us to simulate accurate player data.

III. Applying the Model

A. Empirical Approach

To make sure the data we generate is accurate, we need to make sure there is a way of confirming our results match up with our original inputs. To achieve this, we create a handicap calculator. Given 6-20 rounds of golf from a player, we can apply the USGA's handicap formula and check if our simulation is accurate[4]. Now that we have a way to verify our data, we need to create it.

1) Dividing the Player Handicap: We start by splitting up the player handicap between the 18 holes. The more difficult holes should get more of the handicap, so we give higher-rated holes bigger fractions of the score. Below we can see the arrangement of hole handicaps for the *Denison Golf Club* course which we use to divide up players' course handicaps. We use similar tables for different courses and tees.

Course		**Tees**	**Rating**	**Slope**	**Type**
Denison	Golf Club	Blue	69.8	128	Par
Denison	Golf Club	Blue	69.8	128	Handicap
Hole1	**Hole2**	**Hole3**	**Hole4**	**Hole5**	**Hole6**
4	4	4	3	4	5
17	1	5	13	7	9
Hole7	**Hole8**	**Hole9**	**Hole10**	**Hole11**	**Hole12**
3	4	4	5	4	5
15	3	11	4	8	6
Hole13	**Hole14**	**Hole15**	**Hole16**	**Hole17**	**Hole18**
4	3	4	4	3	4
10	12	2	14	18	16

We start by giving each hole 18 base points. Then, we add (19$-$ hole handicap) points to the base points. This allocates more points to harder holes. This sum is then divided by the total number of points attributed to all of the holes (495), and multiplied by the player's course handicap. We add this to each hole's par and generate a target mean score. Below we can see an example calculation of a 10-Handicap player's target mean score for Hole One on the blue tee. The target score we get is not an integer and is only used to calculate the player's probabilities for the hole.

$$\frac{18 + (19 - 17)}{495} * (10 * \frac{128}{113}) + 4 = 4.457 \quad (5)$$

2) Determining ps and qs: Now that we have a target score, we need to find appropriate p and q values. We plug this target mean score for the hole into a function that works like a binary search algorithm. The algorithm generates a p between a lower and upper bound.

Determining realistic p and q values requires a formula. Looking at this contour plot, we can see that each line represents a different μ score, so many different ps and qs yield the same value. Therefore we have to choose realistic probability values. For example, both points A and B on the graph have μ values of 4.00. However, A is far more reasonable. With point A's probabilities, the player would make good shots 5% of the time, bad shots 4% of the time, and ordinary shots 91% of the time. Professional golfers try to play on par, so this percentage of ordinary strokes still reflects a decent golfer. However, point B's player makes good shots 40% of the time and bad shots 26% of the time, so they are playing erratically in a way that is unrealistic.

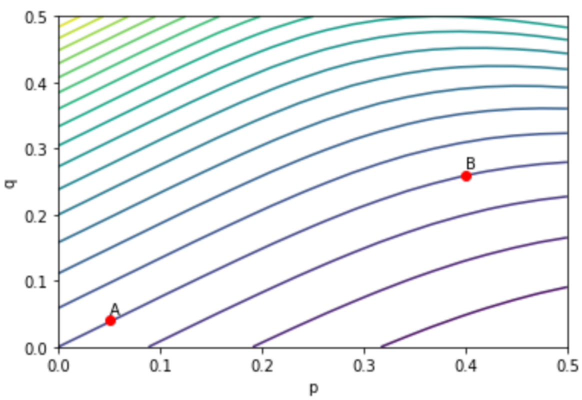

Fig. 2. Contour graph of Mean values with points (.05,.04) and (.4,.26)

To generate reasonable probability values, we consulted a golf expert on what realistic p and q values are for certain mean scores. We plot these values and generate a third-degree polynomial of best fit to them. Point C represents a rookie player that makes a bad shot 40% of the time and an ordinary shot 60% of the time. Contrarily, point D shows a professional player that makes 15% good shots and 85% ordinary shots.

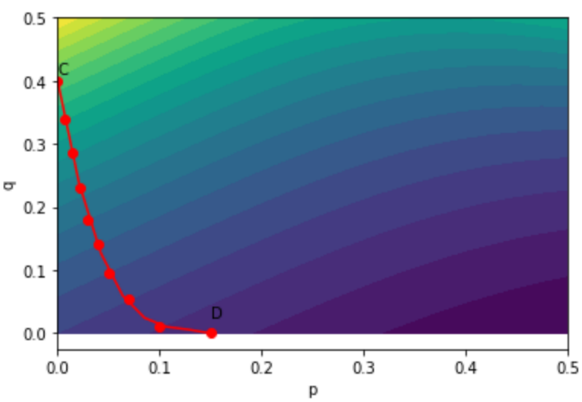

Fig. 3. Contour graph of Mean values

From the graph we get the formula

$$q = 0.404 + -9.208p + 72.067p^2 + -191.032p^3 \qquad (6)$$

With this line, we can find a point on the line to match any given target score. We use the to get a q, which we use in the μ formula, and compare to the μ calculated earlier in a binary search. When the target and μ are close enough, we know we have accurate probability values. Until then, the function will run again with narrowed bounds.

We use these p and q values to simulate a game of golf for the player. After we have the round, we subtract handicap points from the hardest holes until we run out of points. We then compare the net scores of each player and see how many rounds they won. If we want to check their handicap index, we simulate 20 rounds and use the handicap calculator. This practice is useful to check our calculations but, to get conclusive data, we must run 1000 rounds. Since such a simulation is time-consuming, another strategy would be more useful to get decisive data.

B. Analytical Approach

When taking an analytical approach, we use the same beginning framework as in the empirical approach but diverge after calculating the p and q values. We then calculate player probabilities using a probability density function (PDF) and a cumulative density function (CDF). The PDF function uses the probability functions from earlier to calculate the likelihood that the player scores x number of strokes on a hole. The CDF function adds up the probabilities that the player takes from 0 to x strokes to finish the hole. We have a table representing the PDF and CDF values from Hole 1 using player handicaps (PHC) of 10 and 20. These values are the probabilities the players will score from 2 to 7 strokes.

	10	PHC	20	PHC
	PDF	CDF	PDF	CDF
2	.002	.002	.0007	.0007
3	.08	.082	.04	.0407
4	.5	.582	.4	.4407
5	.3	.882	.3	.7407
6	.09	.972	.1	.8407
7	.02	.992	.06	.9007

To compute the probability that a player will win the hole we employ the following method.

```
for i in range(players) do
    WinningProbability = 0
    for j in range(2 to 7) do
        placeholder = PDF(player i scores j strokes on
          this hole)
        for k over players other than i do
            points_Scored = j -
              hole_Handicap_Points(player i) +
              hole_Handicap_Points(player k)
            placeholder = placeholder * (1 -
              CDF(player k scores points_Scored points
              on the hole)
        end
        WinningProbability = WinningProbability +
          placeholder
    end
end
```

The goal of the outer loop is to compute the probability that golfer i wins the hole. Meanwhile, the middle loops computes the probability that player i wins the hole with a real score of j. In the innermost loop we compute the probability that player k loses the hole to player i. This loop uses the points_Scored variable to account for the handicap points each player has applied to the hole. Doing so makes sure player i still has a lower score after using handicaps. Here is an example calculation for calculating the probability of the 10-handicap player winning Hole 1 with 3 strokes. For the sake of simplicity, assume the course handicap is equal to the player handicap. This means that since Hole 1 is the 17-handicap hole, only the 20-handicap player will get a handicap point.

$$.08 * (1 - CDF(20PHC, score = (3 - 0 + 1))) = .08 * (1 - .4407) = .045 \quad (7)$$

The result is added to the probability that the player wins the hole with all other valid number of strokes. This approach requires less processing time and can get useful statistics for each player participating.

IV. Findings

A. Analysis

When applying our analytical algorithm to several varying competitions, we discover that the difference in probabilities varies depending on the number of players. In a two-player game, a player with a handicap advantage on a hole where they receive a stroke advantage has a much higher chance of winning, but a low chance on a hole where the players are playing even (no handicap advantage). If we look at the sums of the probabilities of each player, we can see that players with lower handicaps can expect to win far fewer skins than players with higher handicaps; the ”sum” column represents the expected number of skins each player should win in this match. We demonstrate this with results from a 10 V.S. 20-handicap player game[2].

PHC	Hole1 (17)	Hole2 (1)	Hole3 (5)	Hole4 (13)
10	0.213	0.327	0.525	0.233
20	0.545	0.457	0.233	0.528
Hole5 (7)	Hole6 (9)	Hole7 (15)	Hole8 (3)	Hole9 (11)
0.516	0.511	0.218	0.315	0.494
0.234	0.244	0.541	0.465	0.236
Hole10 (4)	Hole11 (8)	Hole12 (6)	Hole13 (10)	Hole14 (12)
0.316	0.510	0.525	0.499	0.242
0.466	0.235	0.242	0.237	0.522
Hole15 (2)	Hole16 (14)	Hole17 (18)	Hole18 (16)	Sum
0.320	0.237	0.191	0.222	6.414
0.462	0.524	0.566	0.535	7.271

Furthermore, in matches with six golfers, we find the player with the highest handicap wins more than twice the number of skins than the best player with the lowest handicap.

[2] The hole handicaps are in parenthesis.

PHC	Hole1(17)	Hole2 (1)	Hole3 (5)	Hole4 (13)
8	0.023	0.056	0.070	0.011
12	0.018	0.056	0.046	0.008
16	0.014	0.056	0.032	0.136
20	0.165	0.167	0.022	0.103
24	0.132	0.124	0.174	0.079
25	0.125	0.115	0.163	0.074
Hole5 (7)	Hole6 (9)	Hole7 (15)	Hole8 (3)	Hole9 (11)
0.092	0.017	0.009	0.075	0.010
0.062	0.147	0.007	0.049	0.131
0.044	0.108	0.130	0.034	0.096
0.031	0.080	0.099	0.023	0.072
0.022	0.061	0.077	0.170	0.055
0.221	0.057	0.073	0.158	0.051
Hole10 (4)	Hole11 (8)	Hole12 (6)	Hole13 (10)	Hole14 (12)
0.084	0.128	0.078	0.011	0.006
0.057	0.088	0.054	0.134	0.113
0.040	0.062	0.038	0.098	0.081
0.028	0.045	0.027	0.072	0.060
0.164	0.033	0.169	0.055	0.045
0.153	0.030	0.158	0.051	0.042
Hole15 (2)	Hole16 (14)	Hole17 (18)	Hole18 (16)	Sum
0.053	0.016	0.014	0.014	0.746
0.035	0.012	0.011	0.010	1.036
0.023	0.142	0.008	0.138	1.279
0.167	0.110	0.162	0.107	1.541
0.125	0.085	0.129	0.085	1.781
0.116	0.080	0.122	0.080	1.872

To attempt to solve this disparity, we allow the players to use percentages of their handicap instead of the full amount. This means they will not have as many points to apply to the holes. In 4, we graph the correlation between player handicaps and their probability sums against the percent handicap being used. We want there to be no correlation between handicaps and probability sums.

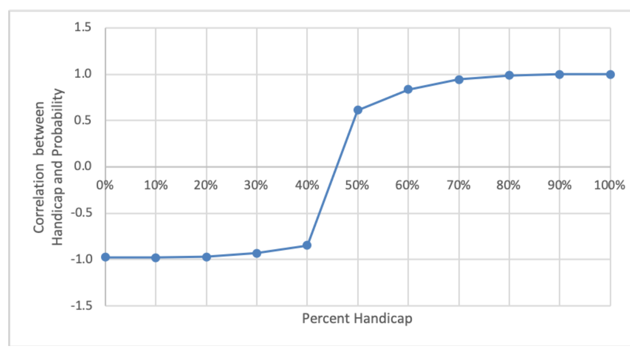

Fig. 4. Contour graph of Mean values

Based on this graph, we try using 45% of the players' handicaps. The players' probability sums are much closer after this change, and the better players have a slight advantage.

PHC	Sum
8	1.438
12	1.275
16	1.203
20	1.187
24	1.260
25	1.179

These probabilities are much more realistic. However, when we apply the same handicap percentage to other games, we do not get such promising results. For example, when we simulate the two-player game from earlier, the new handicap percentage actually causes a worse disparity in wins. Even when we generate another six-player game, the win differences vary.

PHC	Sum
10	7.833
20	5.452

PHC	Sum
2	2.656
4	2.368
6	2.191
8	2.112
10	2.157
12	2.289

B. Conclusions

We can see that skins is clearly inequitable across different skill levels. Players with higher handicaps will on average win far more skins than their lower-handicapped counterparts. In fact, players with lower handicaps statistically do worse than every other player with a higher handicap than them.

We have also found that although using a percentage of player handicaps may seem like a good solution, the method does not work for all combinations of players. Using handicap percentages is useful if the players are close in skill level, but proves less effective if golfers have a wide range of handicaps. If we use this method of equalization, we need to apply a different percentage depending on the players competing. There is no single handicap percentage that works for every game.

V. Future Work

Moving forward, we look for methods to equalize the competition for skins. Based on our current findings, the method may vary based on the number of players competing. Applying 45% of the handicap to each player seems to even the overall odds of the best and worst player, but both have higher chances at winning than players with handicaps between theirs. Furthermore, since our model also only takes three types of shots into account, we worry that it might not accurately capture the nuances of playing golf. We have considered a new model consisting of ”half-shots” of progress, where we still have good, bad, and ordinary shots, but we also have slightly bad and slightly good shots allowing for half a stroke of progress. Overall, we have some promising leads, but have not found any conclusive solutions to making skins equitable across skill levels.

References

[1] Hardy, G. H. “1844. A Mathematical Theorem about Golf.” *The Mathematical Gazette*, vol. 29, no. 287, 1945, pp. 226–227. *JSTOR*, www.jstor.org/stable/3609265. Accessed 9 July 2020.

[2] Van der VEN, A. H. G. S. “The Hardy Distribution for Golf Hole Scores.” *The Mathematical Gazette*, vol. 96, no. 537, 2012, pp. 428–438. *JSTOR*, www.jstor.org/stable/24496865. Accessed 9 July 2020.

[3] OSU Math: https://people.math.osu.edu/husen.1/teaching/571/markov_1.pdf

[4] Golf Software: http://golfsoftware.com/hsd/golf-handicap-formula.html#:~:text=Determine%20a%20course%20handicap%20by,to%20the%20nearest%20whole%20number.&text=The%20following%20assumes%20an%20index,Home%20course%20slope%20of%20120.