Decrypt Betting Behavior

PROBLEM STATEMENT

Bustabit, an online gambling site, has begun awarding users with random bonus multipliers following each successful bet. Bustabit would like to know different sorts of players based on their betting behavior and risk-taking habits in order to better cater to the players, boost the house's income, and leverage a growth in games played by the players.

Online gambling is the act of making bets on risk-reward games in order to gain money. Each person takes bets depending on their own instincts and financial resources. Understanding the similarities and variances in the betting behavior of consumers in this context may assist the gaming platform make educated decisions that can help enhance the platform's income. Lets have a look at few basic rules for playing Bustabit:

1. You place a gamble (in Bits, which is 1 / 1,000,000th of a Bitcoin) and win if you pay out before the game ends.
2. The multiplier value at the time you cashed out determines your win. For example, if you bet 100 and the value at the time you cashed out was 1.25x, you earn 125. Furthermore, a percentage Bonus for every game is multiplied by your bet and added together to offer you your total Profit in a successful game. With a 2% bonus, your profit for this round would be (100 x 1.25) + (100 x0.02) - 100 = 27
3. The multiplier rises with time, but if you wait too long, you may go broke and lose your money.

DATA SET

The following information was obtained from Bustabit for the sake of this study. The collection contains information from 50000 Bustabit games played by 4150 distinct players on the site. The variables in the data are as follows:

1. Id - Unique identifier for a particular row (game result for one player).
2. GameID - Unique identifier for a particular game.
3. Username - Unique identifier for a particular player.
4. Bet - The number of Bits (1 / 1,000,000th of a Bitcoin) bet by the player in this game.
5. CashedOut - The multiplier at which this particular player cashed out.
6. Bonus - The bonus award (in percent) awarded to this player for the game.
7. Profit - The amount this player won in the game, calculated as (Bet * CashedOut) + (Bet * Bonus) – Bet.
8. BustedAt - The multiplier value at which this game busted.
9. PlayDate - The date and time at which this game took place.

The data set consisting of all the above variables looks as follows.

Description: df [6 x 9]

	Id <int>	GameID <int>	Username <chr>	Bet <int>	CashedOut <dbl>	Bonus <dbl>	Profit <dbl>	BustedAt <dbl>	PlayDate <chr>
1	14196549	3366002	papai	5	1.20	0.0	1.00	8.24	2016-11-20T19:44:19Z
2	10676217	3343882	znay22	3	NA	NA	NA	1.40	2016-11-14T14:21:50Z
3	15577107	3374646	rrrrrrrr	4	1.33	3.0	1.44	3.15	2016-11-23T06:39:15Z
4	25732127	3429241	sanya1206	10	NA	NA	NA	1.63	2016-12-08T18:13:55Z
5	17995432	3389174	ADM	50	1.50	1.4	25.70	2.29	2016-11-27T08:14:48Z
6	14147823	3365723	afrod	2	NA	NA	NA	1.04	2016-11-20T17:50:55Z

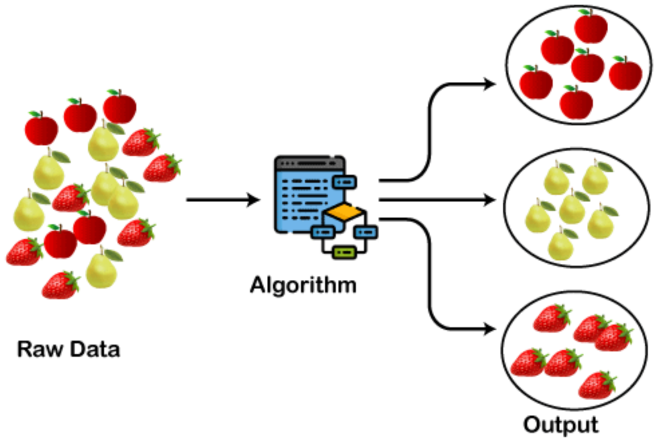
6 rows

Dataset as collected from the source.

Approach

Giving a bigger bonus to cautious players and a lesser bonus to risk takers is not an effective strategy to entice players to wager again. As a result, in order to decide the right amount of bonus to be paid to the players, we use the K-Means clustering technique to separate the players' data into distinct groups depending on their betting behaviour.

K-Means clustering is a popular clustering approach, where K is the number of groups into which the players are separated. The clusters are formed by comparing the similarities and differences (in variables) between observations (game info), and each cluster contains multiple observations that are similar to one another, and each cluster is distinguished from the others based on the differences between observations from one cluster to the next..



Representation of working of K-Means Clustering Algorithm.

In clustering approaches, the similarities between two data can be quantified in terms of distance. More similarities reduce the distance between observations, whereas less similarities increase the distance between them, resulting in the formation of clusters and their centroids. When a new observation is introduced to the cluster, the form of the cluster and hence the centroid observation change until all of the input observations have been split into suitable clusters.

DATA PREPARATION AND APPLICATION OF ALGORITHM

In order to perform the analysis, let us first take a look at how the data looks like

Description: df [50,000 x 9]

Id <int>	GameID <int>	Username <chr>	Bet <int>	CashedOut <dbl>	Bonus <dbl>	Profit <dbl>	BustedAt <dbl>	PlayDate <chr>
19029273	3395044	Shadowshot	130	2.00	2.77	133.60	251025.13	2016-11-29T00:03:05Z
24360604	3421330	bobi3333	200	2.09	1.14	220.28	58615.07	2016-12-06T12:59:14Z
21611265	3407405	gygol	20	1.10	0.00	2.00	26004.72	2016-12-02T13:37:22Z
21611083	3407405	gul	1	20.00	2.00	19.02	26004.72	2016-12-02T13:37:22Z
11048913	3346029	anton02_87	1	6.00	1.00	5.01	20758.70	2016-11-15T04:46:08Z
18161401	3390179	a223532a	20	1.50	1.10	10.22	10650.52	2016-11-27T14:56:37Z
335490	3296540	sofaking	1582	10.02	1.29	14290.04	7278.23	2016-11-01T03:17:03Z
6052058	3322741	darkfreeze2	40	1.50	0.00	20.00	6316.22	2016-11-08T14:09:14Z
5542614	3320673	Wolf84	6	1.67	4.67	4.30	6126.16	2016-11-08T00:00:33Z
18989122	3394816	xrnath	68	1.04	0.00	2.72	4966.31	2016-11-28T22:31:29Z
10913193	3345161	Fossil1554	288	4.16	3.28	919.52	4580.52	2016-11-14T22:59:15Z
10913197	3345161	megainvest	212	3.11	3.28	454.27	4580.52	2016-11-14T22:59:15Z
6193985	3323305	Zoltar328	10	1.04	0.00	0.40	4110.41	2016-11-08T18:01:18Z
6193855	3323305	iluvyou	50	1.56	0.00	28.00	4110.41	2016-11-08T18:01:18Z
9867519	3339674	Dalles	1	1.13	0.00	0.13	4021.92	2016-11-13T09:55:53Z

1-15 of 50,000 rows   Previous 1 2 3 4 5 6 ... 67 Next

Dataset sorted in descending order of BustedAt value.

The following table contains basic information supplied by Bustabit as well as game level data, which were sorted in decreasing order of the value at which a game was busted.

To cluster the players, we must first generate player level statistics from the provided game-wise data. Furthermore, in order to further quantify player behavior, we will require additional variables such as losses suffered when a game was busted and an indicator variable that quantifies whether the game was a win or loss. To get per-player statistics, we average the values of all the new data variables for each player. Following the completion of all of the above data cleaning and preparation procedures, the final input data including per-player metrics would be as follows.

With the standardized data of per-players features, we now apply the K-Means clustering algorithm in order to cluster the players based on their gambling behaviour. Upon applying the algorithm, the following were the number of observations or players sorted into 5 different clusters.

Representation of Input data with Average Stats per Player

```
# Displaying the cleaned data
head(player_data_Clustering)
```

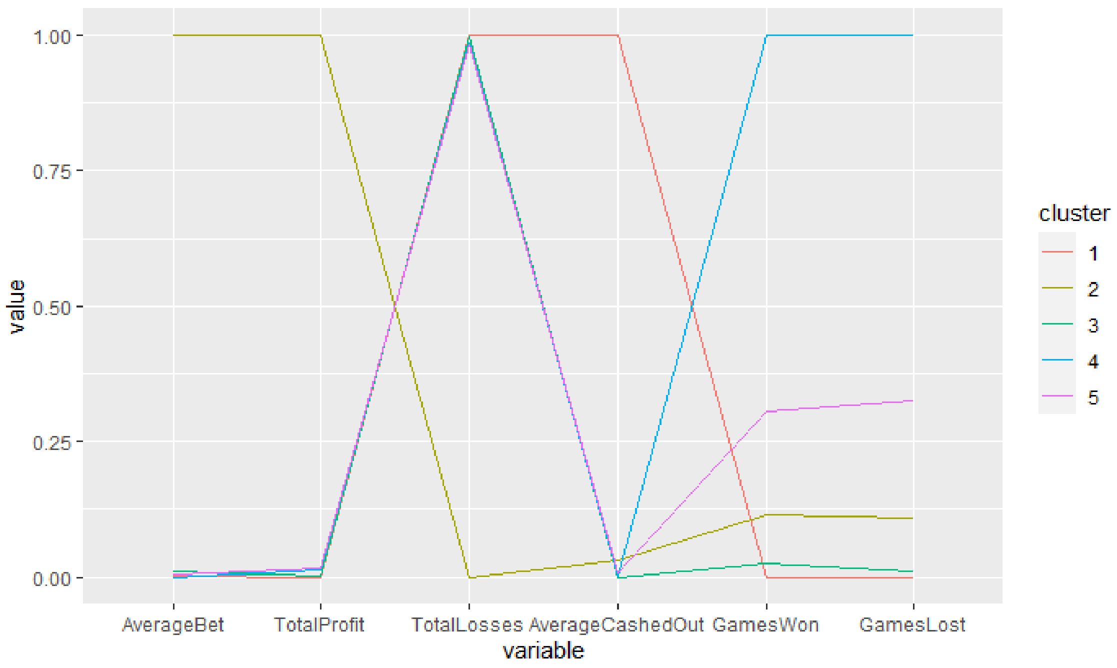
```
## # A tibble: 6 x 7
##   Username     AverageCashedOut AverageBet TotalProfit TotalLo~1 Games~2 Games~3
##   <chr>                   <dbl>      <dbl>       <dbl>     <dbl>   <dbl>   <dbl>
## 1 _caramba_tm_             1.7        1.33        3.13         0       3       0
## 2 _Dear_                   1.66     215           0         -860       0       4
## 3 _lsx                     1.20    6282        3545.       -2000       4       1
## 4 _noBap_                  6.58       4           0           -4       0       1
## 5 _TechDeck                1.19       6           0           -6       0       1
## 6 -__---                   1.33   21776.     183322.     -116046      19       5
## # ... with abbreviated variable names 1: TotalLosses, 2: GamesWon, 3: GamesLost
```

```
##
##    1    2    3    4    5
## 3626   17   16   78  412
```

```
# Group by the cluster assignment and calculate averages
cluster_avg <- player_data_Clustering %>%
    group_by(cluster) %>%
```

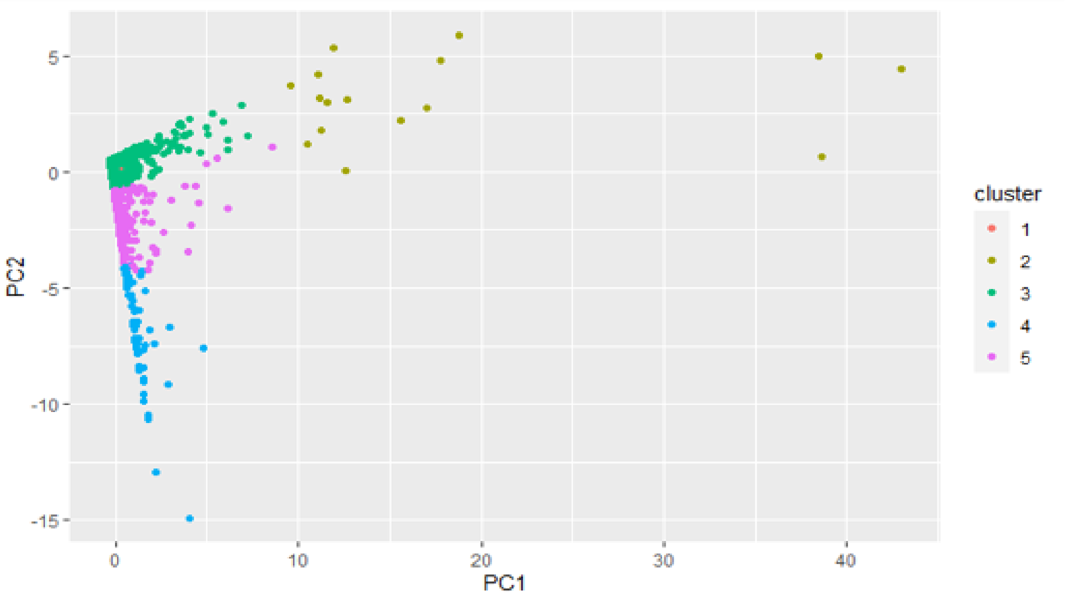
Number of Players sorted into 5 distinct clusters.

In order to clearly distinguish the clusters from one another, we compute the average values of the attributes in each cluster and compare them. The following is the final result of the clusters and their average values in each attribute which helps us determine the distinct features of each cluster based on their gambling behaviour.



The graph above depicts the values of each cluster's variables on a scale of 0 to 1, distinguishing one cluster from another, with each cluster having distinct values across the variables. A number of 1 indicates that a cluster has the highest value compared to all other clusters, while a value of 0 suggests that it has the lowest.

To comprehend the plot in two dimensions, the clusters can alternatively be represented as a scatter plot of players plotted against two of the major components. The scatter plot of the players separated into clusters with color coding is shown in the graph below.



Scatter plot of players with principal components.

can be differentiated looking at the placement of the observations of each cluster in the graph.

OUTPUT ANALYSIS

The application of K-Means clustering algorithm to the per-player input data gave us the following output based on the gambling behaviour of players.

```
## # A tibble: 5 x 7
##   cluster AverageCashedOut AverageBet TotalProfit TotalLosses GamesWon GamesLost
##   <fct>              <dbl>      <dbl>       <dbl>       <dbl>    <dbl>     <dbl>
## 1 1                   1.70      4024.       4273.      -4366.    2.91       2.13
## 2 2                  27.4       1278.        619.       -581.    0.706      1.53
## 3 3                   2.47    298946.    1198191.  -1056062.   10.6        8.06
## 4 4                   1.76       432.      18568.     -16724.   87.2       61.2
## 5 5                   1.92      1633.      19363.     -19205.   27.1       21.0
```

3

Output of K-Means Clustering along with Labels

The above clusters can be interpreted as below based on the distinct values of each variable among the clusters. Each cluster is named appropriately based on the attributes of the clusters.

Cautious Commoners:

This is the largest of the five clusters, and might be described as the more casual Bustabit players. They've played the fewest number of games overall, and tend to make more conservative bets in general.

Strategic Addicts:

These users play a lot of games on Bustabit, but tend to keep their bets under control. As a result, they've made on average net positive earnings from the site, in spite of having the most games played. They seem to maintain a strategy (or an automated script/bot) that works to earn them money.

Risky Commoners:

These users seem to be a step above the Cautious Commoners in their Bustabit gambling habits, making larger average bets, and playing a larger number of games on the site. As a result, though they have about the same number of average games won as the Risk Takers, they have a significantly higher number of games lost.

Risk Takers:

These users have played only a couple games on average, but their average cashed out value is significantly higher than the other clusters, indicating that they tend to wait for the multiplier to increase to large values before cashing out.

High Rollers:

High bets are the name of the game for this group. They bet large sums of money in each game, although they tend to cash out at lower multipliers and thus play the game more conservatively, particularly compared to the Risk Takers. Interestingly, these users have also on average earned net positive earnings from their games played.

CONCLUSION

Based on the above analysis of Bustabit's data and clustering based on player gambling behavior, we can conclude that the online gambling platform will now be able to make informed decisions on the amount of bonus to be given to different clusters of players in order to leverage increased revenue with increased customer game play. For example, a player in the 'Strategic Addicts' cluster would be more likely to play more games and increase his wager if he noticed a trend of bonus being doubled every third game he played.

As a result, K-Means clustering is extremely useful for gambling platforms in making strategic judgments on how to advance their marketing strategies in order to increase revenue and profits.